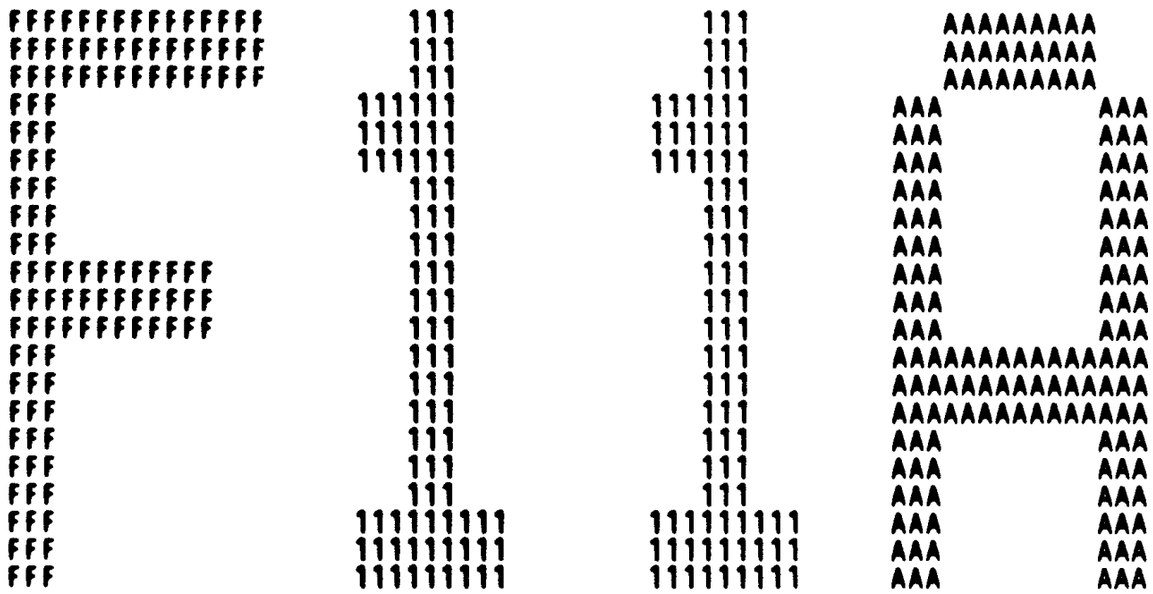
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
111111111
111111111
111111111
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
111111111
111111111
111111111
AAAAAAAAA
AAAAAAAAA
AAAAAAAAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA

```
RRRRRRRR    WW      WW    AAAAAA    TTTTTTTTTT  TTTTTTTTTT  RRRRRRRR
RRRRRRRR    WW      WW    AAAAAA    TTTTTTTTTT  TTTTTTTTTT  RRRRRRRR
RR      RR  WW      WW   AA    AA       TT          TT      RR      RR
RR      RR  WW      WW   AA    AA       TT          TT      RR      RR
RR      RR  WW      WW   AA    AA       TT          TT      RR      RR
RR      RR  WW      WW   AA    AA       TT          TT      RR      RR
RRRRRRRR    WW      WW   AA    AA       TT          TT      RRRRRRRR
RRRRRRRR    WW      WW   AA    AA       TT          TT      RRRRRRRR
RR  RR      WW  WW  WW   AAAAAAAAAA     TT          TT      RR  RR
RR  RR      WW  WW  WW   AAAAAAAAAA     TT          TT      RR  RR
RR    RR    WWWW  WWWW   AA    AA       TT          TT      RR    RR        ....
RR    RR    WWWW  WWWW   AA    AA       TT          TT      RR    RR        ....
RR      RR  WW      WW   AA    AA       TT          TT      RR      RR      ....
RR      RR  WW      WW   AA    AA       TT          TT      RR      RR      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
;     1          0001  0 MODULE RWATTR (
;     2          0002  0                 LANGUAGE (BLISS32),
;     3          0003  0                 IDENT = 'V04-000'
;     4          0004  0                 ) =
;     5          0005  1 BEGIN
;     6          0006  1
;     7          0007  1 !
;     8          0008  1 !**********************************************************************
;     9          0009  1 !*                                                                    *
;    10          0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                           *
;    11          0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.            *
;    12          0012  1 !*  ALL RIGHTS RESERVED.                                             *
;    13          0013  1 !*                                                                    *
;    14          0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
;    15          0015  1 !*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE   *
;    16          0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
;    17          0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
;    18          0018  1 !*  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
;    19          0019  1 !*  TRANSFERRED.                                                     *
;    20          0020  1 !*                                                                    *
;    21          0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
;    22          0022  1 !*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
;    23          0023  1 !*  CORPORATION.                                                     *
;    24          0024  1 !*                                                                    *
;    25          0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
;    26          0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.          *
;    27          0027  1 !*                                                                    *
;    28          0028  1 !*                                                                    *
;    29          0029  1 !**********************************************************************
;    30          0030  1
;    31          0031  1 !++
;    32          0032  1 !
;    33          0033  1 ! FACILITY:  F11ACP Structure Level 1
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !
;    37          0037  1 !       This module contains the code and tables to process the read
;    38          0038  1 !       and write attributes functions.
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:
;    41          0041  1 !
;    42          0042  1 !       STARLET operating system, including privileged system services
;    43          0043  1 !       and internal exec routines. This routine must be executed
;    44          0044  1 !       in kernel mode.
;    45          0045  1 !
;    46          0046  1 !--
;    47          0047  1 !
;    48          0048  1 !
;    49          0049  1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  6-Jan-1977  21:05
;    50          0050  1 !
;    51          0051  1 ! MODIFIED BY:
;    52          0052  1 !
;    53          0053  1 !       V03-008 LMP0241         L. Mark Pilant,         26-Apr-1984  10:47
;    54          0054  1 !               Include the FIB in the MAKE_NAMEBLOCK routine call.
;    55          0055  1 !
;    56          0056  1 !       V03-007 ACG0415         Andrew C. Goldstein,    9-Apr-1984  16:26
;    57          0057  1 !               Fix probing of access mode ATR's; extend ascii file name to
```

```
;    58          0058  1 !                       86 bytes
;    59          0059  1 !
;    60          0060  1 !               V03-006 ACG0410         Andrew C. Goldstein,    22-Mar-1984  22:31
;    61          0061  1 !                       Add support for access mode attribute
;    62          0062  1 !
;    63          0063  1 !               V03-005 LMP0154         L. Mark Pilant,         14-Sep-1983  11:21
;    64          0064  1 !                       Bring the attribute table up to the same level as the
;    65          0065  1 !                       attribute definition macro ($ATRDEF).
;    66          0066  1 !
;    67          0067  1 !               V03-004 ACG0355         Andrew C. Goldstein,    30-Aug-1983  15:21
;    68          0068  1 !                       Fix bug in high order UIC test in ACG0329
;    69          0069  1 !
;    70          0070  1 !               V03-003 ACG0329         Andrew C. Goldstein,    12-Apr-1983  14:18
;    71          0071  1 !                       Fold long UIC's into [377,377] for 16 bit UIC
;    72          0072  1 !
;    73          0073  1 !               V03-002 STJ3079         Steven T. Jeffreys,     26-Mar-1983
;    74          0074  1 !                       Add dummy HIGHWATER attribute.
;    75          0075  1 !
;    76          0076  1 !               V03-001 LMP0066         L. Mark Pilant,         15-Dec-1982  10:04
;    77          0077  1 !                       Handle ACL attributes gracefully rather than blowing away
;    78          0078  1 !                       with a bad attribute error.
;    79          0079  1 !
;    80          0080  1 !               V02-008 ACG0253         Andrew C. Goldstein,    18-Jan-1982  16:23
;    81          0081  1 !                       Add dummy HDR1 accessibility attribute
;    82          0082  1 !
;    83          0083  1 !               V02-007 ACG0232         Andrew C. Goldstein,    4-Dec-1981  16:47
;    84          0084  1 !                       Protect HIBLK during write attributes
;    85          0085  1 !
;    86          0086  1 !               V02-006 ACG0229         Andrew C. Goldstein,    1-Dec-1981  13:40
;    87          0087  1 !                       Extend statistics block to include full counts
;    88          0088  1 !
;    89          0089  1 !               V02-005 ACG0221         Andrew C. Goldstein,    30-Oct-1981  18:07
;    90          0090  1 !                       Add file attribute for journal flags
;    91          0091  1 !
;    92          0092  1 !               V02-004 MLJ0029         Martin L. Jack, 11-Jul-1981  23:40
;    93          0093  1 !                       Clean up file header to quadword time conversion so that
;    94          0094  1 !                       hundredths of seconds are taken as zero rather than
;    95          0095  1 !                       propagating from current time, and ensure that zero is
;    96          0096  1 !                       returned for an invalid time string.
;    97          0097  1 !
;    98          0098  1 !               A0103   ACG0097         Andrew C. Goldstein,    18-Dec-1979  19:15
;    99          0099  1 !                       Protect SPOOL, BAD, DELETE, etc., file char's
;   100          0100  1 !
;   101          0101  1 !               A0102   ACG0093         Andrew C. Goldstein,    6-Dec-1979  19:58
;   102          0102  1 !                       Add dummy backlink attribute, fix user char format
;   103          0103  1 !
;   104          0104  1 !               A0101   ACG0023         Andrew C. Goldstein,  19-Feb-1979  13:23
;   105          0105  1 !               Fix max attribute code to include DIRSEQ
;   106          0106  1 !
;   107          0107  1 !               A0100   ACG00001        Andrew C. Goldstein,  10-Oct-1978  20:03
;   108          0108  1 !               Previous revision history moved to F11A.REV
;   109          0109  1 !**
;   110          0110  1
;   111          0111  1
;   112          0112  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;   113          0113  1 REQUIRE 'SRC$:FCPDEF.B32';
;   114          0428  1
```

```
;  115          0429  1
;  116          0430  1 FORWARD ROUTINE
;  117          0431  1         READ_ATTRIB,                    ! read attributes
;  118          0432  1         FID_TO_SPEC     : NOVALUE,      ! convert FID to file-spec
;  119          0433  1         WRITE_ATTRIB    : NOVALUE;      ! write attributes
```

```
;  121     0434  1 !++
;  122     0435  1 !
;  123     0436  1 ! Attribute control table. The table is indexed by attribute number.
;  124     0437  1 ! The first byte contains random flags.  The second byte contains
;  125     0438  1 ! the attribute location code.  The third byte contains the offset
;  126     0439  1 ! of the attribute from its location origin. The fourth byte contains
;  127     0440  1 ! the index of the action routine to process the attribute.  The fifth
;  128     0441  1 ! and sixth bytes (a word value) contains the maximum size of the
;  129     0442  1 ! attribute.
;  130     0443  1 !
;  131     0444  1 !--
;  132     0445  1
;  133     0446  1 ! Macros and literals to access the table entries.
;  134     0447  1 !
;  135     0448  1
;  136     0449  1 MACRO
;  137     0450  1         ATC_READ_ONLY   = 0,0,1,0%,     ! read only attribute
;  138     0451  1         ATC_PROTECTED   = 0,1,1,0%,     ! writable by file owner only
;  139     0452  1         ATC_LOCKED      = 0,2,1,0%,     ! subject to file access locks
;  140     0453  1         ATC_LOCATION    = 1,0,8,0%,     ! 3 bit location code
;  141     0454  1         ATC_OFFSET      = 2,0,8,0%,     ! location offset
;  142     0455  1         ATC_ACTION      = 3,0,8,0%,     ! action routine
;  143     0456  1         ATC_MAX_SIZE    = 4,0,16,0%;    ! max attribute size
;  144     0457  1
;  145     0458  1
;  146     0459  1 ! Masks for the flags.
;  147     0460  1 !
;  148     0461  1
;  149     0462  1 LITERAL
;  150     0463  1         M_READ_ONLY     = 1,
;  151     0464  1         M_PROTECTED     = 2,
;  152     0465  1         M_LOCKED        = 4;
;  153     0466  1
;  154     0467  1
;  155     0468  1 ! Attribute location codes.
;  156     0469  1 !
;  157     0470  1
;  158     0471  1 LITERAL
;  159     0472  1         ATC_ZERO        = 0,            ! zero - no location
;  160     0473  1         ATC_FCB         = 1,            ! in file control block
;  161     0474  1         ATC_HEADER      = 2,            ! file header header area
;  162     0475  1         ATC_IDENT       = 3,            ! file header ident area
;  163     0476  1         ATC_MAP         = 4,            ! file header map area
;  164     0477  1         ATC_ACL         = 5,            ! file header Access Control List area
;  165     0478  1         ATC_ACPGBL      = 6,            ! ACP global storage
;  166     0479  1         ATC_FID2NAME    = 7,            ! Convert FID to file spec
;  167     0480  1
;  168     0481  1         ATC_LASTATC     = 7;            ! last location code
;  169     0482  1
;  170     0483  1
;  171     0484  1 ! ACP global storage index codes.
;  172     0485  1 !
;  173     0486  1
;  174     0487  1 LITERAL
;  175     0488  1         GBL_PRV         = 0,            ! Privileges used to gain access
;  176     0489  1         GBL_ACE         = 1,            ! ACE used to gain access
;  177     0490  1
```

```
;  178          0491  1          GBL_LASTGBL     = 1;            . Last index code
;  179          0492  1
;  180          0493  1
;  181          0494  1 ! Attribute processing action routines.
;  182          0495  1 !
;  183          0496  1
;  184          0497  1 LITERAL
;  185          0498  1          ACT_NOP         = 0,            ! ignore attribute
;  186          0499  1          ACT_ILLEGAL     = 1,            ! illegal attribute code
;  187          0500  1          ACT_COPY        = 2,            ! simple copy
;  188          0501  1          ACT_STATBLK     = 3,            ! build statistics block
;  189          0502  1          ACT_ZERO        = 4,            ! zero valued attribute
;  190          0503  1          ACT_BLOCKSIZE   = 5,            ! medium block size
;  191          0504  1          ACT_ASCNAME     = 6,            ! ASCII file name
;  192          0505  1          ACT_DATIME      = 7,            ! 64 bit date and time
;  193          0506  1          ACT_UIC         = 8,            ! 4 byte file owner UIC
;  194          0507  1          ACT_UCHAR       = 9,            ! user controlled characteristics
;  195          0508  1          ACT_BLANK       = 10,           ! blank character attribute
;  196          0509  1          ACT_ACL         = 11,           ! Access Control List
;  197          0510  1          ACT_ACMODE      = 12,           ! buffer access mode
;  198          0511  1
;  199          0512  1          ACT_LASTACT     = 12;           ! highest action routine code
;  200          0513  1
;  201          0514  1
;  202          0515  1 ! Macro to build table entry.
;  203          0516  1 !
;  204          0517  1
;  205          0518  1 MACRO
;  206        M 0519  1          ATTRIBUTE (CODE, FLAGS, LOC, OFF1, OFF2, OFF3, OFF4, SIZE, ACTION) =
;  207        M 0520  1                  BYTE    (FLAGS,
;  208        M 0521  1                           LOC,
;  209        M 0522  1                           $BYTEOFFSET (OFF1, OFF2, OFF3, OFF4),
;  210        M 0523  1                           ACTION),
;  211        M 0524  1                  WORD    (SIZE,
;  212        M 0525  1                           0)
;  213          0526  1                  %;
;  214          0527  1
;  215          0528  1 MACRO
;  216          0529  1          NULL_FIELD      = 0,0,0,0%;
;  217          0530  1
;  218          0531  1
;  219          0532  1 ! The attribute control table itself.
;  220          0533  1 !
;  221          0534  1
;  222          0535  1 BIND
;  223          0536  1          ATC     = UPLIT BYTE (
;  224          0537  1
;  225          0538  1 ATTRIBUTE (0,                   M_PROTECTED,    ATC_HEADER,     FH1$W_FILEOWNER,5,      ACT_COPY),
;  226          0539  1 ATTRIBUTE (0,                   M_PROTECTED,    ATC_HEADER,     FH1$W_FILEPROT, 3,      ACT_COPY),
;  227          0540  1 ATTRIBUTE (ATR$C_UCHAR,         M_LOCKED,       ATC_HEADER,     FH1$B_USERCHAR, 4,      ACT_UCHAR),
;  228          0541  1 ATTRIBUTE (ATR$C_RECATTR,       M_LOCKED,       ATC_HEADER,     FH1$W_RECATTR,  32,     ACT_COPY),
;  229          0542  1 ATTRIBUTE (ATR$C_FILNAM,        0,              ATC_IDENT,      FI1$W_FILENAME, 10,     ACT_COPY),
;  230          0543  1 ATTRIBUTE (ATR$C_FILTYP,        0,              ATC_IDENT,      FI1$W_FILETYPE, 4,      ACT_COPY),
;  231          0544  1 ATTRIBUTE (ATR$C_FILVER,        0,              ATC_IDENT,      FI1$W_VERSION,  2,      ACT_COPY),
;  232          0545  1 ATTRIBUTE (ATR$C_EXPDAT,        M_PROTECTED,    ATC_IDENT,      FI1$T_EXPDATE,  7,      ACT_COPY),
;  233          0546  1 ATTRIBUTE (ATR$C_STATBLK,       M_READ_ONLY,    ATC_FCB,        NULL_FIELD,     32,     ACT_STATBLK),
;  234          0547  1 ATTRIBUTE (ATR$C_HEADER,        M_READ_ONLY,    ATC_HEADER,     NULL_FIELD,     512,    ACT_COPY),
```

```
;  235     0548  1 ATTRIBUTE (ATR$C_BLOCKSIZE,  0,              ATC_ZERO,      NULL_FIELD,      2,    ACT_BLOCKSIZE),
;  236     0549  1 ATTRIBUTE (ATR$C_USERLABEL,  0,              ATC_ZERO,      NULL_FIELD,      80,   ACT_ZERO),
;  237     0550  1 ATTRIBUTE (ATR$C_ASCDATES,   M_PROTECTED,    ATC_IDENT,     FI1$Q_REVISION,  35,   ACT_COPY),
;  238     0551  1 ATTRIBUTE (ATR$C_ALCONTROL,  0,              ATC_ZERO,      NULL_FIELD,      14,   ACT_NOP),
;  239     0552  1 ATTRIBUTE (ATR$C_ENDLBLAST,  0,              ATC_ZERO,      NULL_FIELD,      1,    ACT_NOP),
;  240     0553  1 ATTRIBUTE (ATR$C_ASCNAME,    0,              ATC_IDENT,     FI1$Q_FILENAME,  86,   ACT_ASCNAME),
;  241     0554  1 ATTRIBUTE (ATR$C_CREDATE,    M_PROTECTED,    ATC_IDENT,     FI1$T_CREDATE,   8,    ACT_DATIME),
;  242     0555  1 ATTRIBUTE (ATR$C_REVDATE,    M_PROTECTED,    ATC_IDENT,     FI1$T_REVDATE,   8,    ACT_DATIME),
;  243     0556  1 ATTRIBUTE (ATR$C_EXPDATE,    M_PROTECTED,    ATC_IDENT,     FI1$T_EXPDATE,   8,    ACT_DATIME),
;  244     0557  1 ATTRIBUTE (ATR$C_BAKDATE,    M_PROTECTED,    ATC_ZERO,      NULL_FIELD,      8,    ACT_ZERO),
;  245     0558  1 ATTRIBUTE (ATR$C_UIC,        M_PROTECTED,    ATC_HEADER,    FH1$Q_FILEOWNER, 4,    ACT_UIC),
;  246     0559  1 ATTRIBUTE (ATR$C_FPRO,       M_PROTECTED,    ATC_HEADER,    FH1$W_FILEPROT,  2,    ACT_COPY),
;  247     0560  1 ATTRIBUTE (ATR$C_RPRO,       M_PROTECTED,    ATC_ZERO,      NULL_FIELD,      2,    ACT_ZERO),
;  248     0561  1 ATTRIBUTE (ATR$C_ACLEVEL,    M_PROTECTED,    ATC_ZERO,      NULL_FIELD,      2,    ACT_ZERO),
;  249     0562  1 ATTRIBUTE (ATR$C_SEMASK,     M_PROTECTED,    ATC_ZERO,      NULL_FIELD,      8,    ACT_ZERO),
;  250     0563  1 ATTRIBUTE (ATR$C_UIC_RO,     M_READ_ONLY,    ATC_HEADER,    FH1$Q_FILEOWNER, 4,    ACT_UIC),
;  251     0564  1 ATTRIBUTE (ATR$C_DIRSEQ,     M_READ_ONLY,    ATC_ZERO,      NULL_FIELD,      2,    ACT_ZERO),
;  252     0565  1 ATTRIBUTE (ATR$C_BACKLINK,   M_PROTECTED,    ATC_ZERO,      NULL_FIELD,      6,    ACT_ZERO),
;  253     0566  1 ATTRIBUTE (ATR$C_JOURNAL,    M_PROTECTED,    ATC_ZERO,      NULL_FIELD,      2,    ACT_ZERO),
;  254     0567  1 ATTRIBUTE (ATR$C_HDR1_ACC,   M_PROTECTED,    ATC_ZERO,      NULL_FIELD,      1,    ACT_BLANK),
;  255     0568  1 ATTRIBUTE (ATR$C_ADDACLENT,  M_LOCKED,       ATC_ZERO,      NULL_FIELD,      255,  ACT_ACL),
;  256     0569  1 ATTRIBUTE (ATR$C_DELACLENT,  M_LOCKED,       ATC_ACL,       NULL_FIELD,      255,  ACT_ACL),
;  257     0570  1 ATTRIBUTE (ATR$C_MODACLENT,  M_LOCKED,       ATC_ACL,       NULL_FIELD,      255,  ACT_ACL),
;  258     0571  1 ATTRIBUTE (ATR$C_FNDACLENT,  M_READ_ONLY,    ATC_ACL,       NULL_FIELD,      255,  ACT_ACL),
;  259     0572  1 ATTRIBUTE (ATR$C_FNDACETYP,  M_READ_ONLY,    ATC_ACL,       NULL_FIELD,      255,  ACT_ACL),
;  260     0573  1 ATTRIBUTE (ATR$C_DELETEACL,  M_LOCKED,       ATC_ACL,       NULL_FIELD,      255,  ACT_ACL),
;  261     0574  1 ATTRIBUTE (ATR$C_READACL,    M_READ_ONLY,    ATC_ACL,       NULL_FIELD,      512,  ACT_ACL),
;  262     0575  1 ATTRIBUTE (ATR$C_ACLLENGTH,  M_READ_ONLY,    ATC_ACL,       NULL_FIELD,      4,    ACT_ACL),
;  263     0576  1 ATTRIBUTE (ATR$C_READACE,    M_READ_ONLY,    ATC_ACL,       NULL_FIELD,      255,  ACT_ACL),
;  264     0577  1 ATTRIBUTE (ATR$C_RESERVED,   M_PROTECTED,    ATC_ZERO,      NULL_FIELD,      380,  ACT_ZERO),
;  265     0578  1 ATTRIBUTE (ATR$C_HIGHWATER,  M_READ_ONLY,    ATC_HEADER,    NULL_FIELD,      4,    ACT_ZERO),
;  266     0579  1 ATTRIBUTE (ATR$C_ACCESS_MASK, M_READ_ONLY,   ATC_ACPGBL,    NULL_FIELD,      4,    ACT_ILLEGAL),
;  267     0580  1 ATTRIBUTE (ATR$C_PRIVS_USED, M_READ_ONLY,    ATC_ACPGBL,    GBL_PRV,0,0,0,   4,    ACT_COPY),
;  268     0581  1 ATTRIBUTE (ATR$C_MATCHING_ACE, M_READ_ONLY,  ATC_ACPGBL,    GBL_ACE,0,0,0,   255,  ACT_ZERO),
;  269     0582  1 ATTRIBUTE (ATR$C_ACCESS_MODE, M_READ_ONLY,   ATC_ZERO,      NULL_FIELD,      1,    ACT_ACMODE),
;  270     0583  1 ATTRIBUTE (ATR$C_FILE_SPEC,  M_READ_ONLY,    ATC_FID2NAME,  NULL_FIELD,      512,  ACT_COPY),
;  271     0584  1 ATTRIBUTE (ATR$C_CLASS_MASK, 0,              ATC_ZERO,      NULL_FIELD,      0,    ACT_ZERO),
;  272     0585  1 ATTRIBUTE (ATR$C_BUFFER_OFFSET, 0,           ATC_ZERO,      NULL_FIELD,      0,    ACT_NOP)
;  273     0586  1
;  274     0587  1                 ) : BBLOCKVECTOR [,8];
;  275     0588  1
;  276     0589  1 LITERAL
;  277     0590  1         MAX_CODE        = ATR$C_MAX_CODE; ! highest attribute code
;  278     0591  1
;  279     0592  1
;  280     0593  1 ! Storage for the converted file specification.
;  281     0594  1
;  282     0595  1 OWN
;  283     0596  1
;  284     0597  1 ! ***** The following must be adjacent. *****
;  285     0598  1
;  286     0599  1         FILE_SPEC_LEN   : VECTOR [1,WORD] ALIGN (0),    ! File spec length
;  287     0600  1         FULL_FILE_SPEC  : VECTOR [510,BYTE] ALIGN (0);  ! File spec storage
;  288     0601  1
;  289     0602  1 ! ***** The preceeding must be adjacent. *****
;  290     0603  1
;  291     0604  1 !
```

```
;  292          0605  1 ! Protected bits in the file characteristics longword. These may not be
;  293          0606  1 ! modified by write attributes calls.
;  294          0607  1 !
;  295          0608  1 LITERAL
;  296          0609  2         PROTECTED_CHAR  =  $FIELDMASK (FH1$V_CONTIG)
;  297          0610  1                         OR $FIELDMASK (FH1$V_SPOOL)^8
;  298          0611  1                         OR $FIELDMASK (FH1$V_BADBLOCK)^8
;  299          0612  1                         OR $FIELDMASK (FH1$V_MARKDEL)^8;
```

```
;  301          0613  1 GLOBAL ROUTINE READ_ATTRIB (HEADER, ABD) =
;  302          0614  1
;  303          0615  1 !++
;  304          0616  1 !
;  305          0617  1 ! FUNCTIONAL DESCRIPTION:
;  306          0618  1 !
;  307          0619  1 !       This routine performs the read attributes function. The
;  308          0620  1 !       requested attributes are assembled into the buffer packet.
;  309          0621  1 !
;  310          0622  1 ! CALLING SEQUENCE:
;  311          0623  1 !       READ_ATTRIB (ARG1, ARG2)
;  312          0624  1 !
;  313          0625  1 ! INPUT PARAMETERS:
;  314          0626  1 !       ARG1: address of file header
;  315          0627  1 !       ARG2: address of buffer descriptors
;  316          0628  1 !
;  317          0629  1 ! IMPLICIT INPUTS:
;  318          0630  1 !       IO_PACKET: I/O packet for this operation
;  319          0631  1 !       PRIMARY_FCB: FCB of file
;  320          0632  1 !
;  321          0633  1 ! OUTPUT PARAMETERS:
;  322          0634  1 !       ARG2: address of buffer descriptors
;  323          0635  1 !
;  324          0636  1 ! IMPLICIT OUTPUTS:
;  325          0637  1 !       NONE
;  326          0638  1 !
;  327          0639  1 ! ROUTINE VALUE:
;  328          0640  1 !       1 if successful
;  329          0641  1 !       0 if error
;  330          0642  1 !
;  331          0643  1 ! SIDE EFFECTS:
;  332          0644  1 !       attribute data written into buffer packet
;  333          0645  1 !
;  334          0646  1 !--
;  335          0647  1
;  336          0648  2 BEGIN
;  337          0649  2
;  338          0650  2 MAP
;  339          0651  2         HEADER          : REF BBLOCK,   ! file header arg
;  340          0652  2         ABD             : REF BBLOCKVECTOR [,ABD$C_LENGTH];
;  341          0653  2                                         ! buffer descriptor arg
;  342          0654  2
;  343          0655  2 LITERAL
;  344          0656  2         DATLEN          = 23;           ! length of date string buffer
;  345          0657  2
;  346          0658  2 LOCAL
;  347          0659  2         LOCAL_HEADER    : REF BBLOCK,   ! Local copy of header address
;  348          0660  2         ACCESS_MODE,                    ! access mode to set for attribute buffer
;  349          0661  2         P,                              ! pointer to attribute text
;  350          0662  2         COUNT,                          ! attribute size desired
;  351          0663  2         ADDRESS         : REF BBLOCK,   ! address of attribute
;  352          0664  2         CODE,                           ! attribute code
;  353          0665  2         MAX_COUNT,                      ! max size of attribute
;  354          0666  2         DATBUF          : VECTOR [DATLEN, BYTE], ! buffer to build date string
;  355          0667  2         DATDESC         : VECTOR [2],   ! string descriptor for date string
;  356          0668  2         ACTION          : BYTE,         ! code of action routine
;  357          0669  2         ATT_BUFFER      : BBLOCK [86],  ! buffer to build reformatted attribute text
```

```
;  358     0670  2          ACE_POINTER     : REF BBLOCK,   ! address of user ACE
;  359     0671  2          STATUS;                         ! routine exit status
;  360     0672  2
;  361     0673  2 EXTERNAL
;  362     0674  2          IO_PACKET       : REF BBLOCK,   ! I/O packet in process
;  363     0675  2          PRIMARY_FCB     : REF BBLOCK,   ! FCB of file
;  364     0676  2          CURRENT_WINDOW  : REF BBLOCK,   ! window of file
;  365     0677  2          FILE_HEADER     : REF BBLOCK;   ! address of current file header
;  366     0678  2
;  367     0679  2 EXTERNAL ROUTINE
;  368     0680  2          PMS_START_SUB,                  ! start subfunction metering
;  369     0681  2          PMS_END_SUB,                    ! end subfunction metering
;  370     0682  2          MAKE_STRING,                    ! convert file name block to string
;  371     0683  2          SYS$BINTIM      : ADDRESSING_MODE (ABSOLUTE),
;  372     0684  2                                          ! convert string to binary time
;  373     0685  2          READ_HEADER;                    ! Read a file header
;  374     0686  2
;  375     0687  2
;  376     0688  2 ! Start metering for this subfunction.
;  377     0689  2 !
;  378     0690  2
;  379     0691  2 PMS_START_SUB (PMS_RWATT);
;  380     0692  2
;  381     0693  2 STATUS = 1;                             ! assume success
;  382     0694  2 LOCAL_HEADER = .HEADER;         ! Copy header address
;  383     0695  2
;  384     0696  2 ! Set the buffered read bit in the I/O packet to indicate to IO_DONE that
;  385     0697  2 ! the attribute buffers are valid.
;  386     0698  2 !
;  387     0699  2
;  388     0700  2 IO_PACKET[IRP$V_FUNC] = 1;
;  389     0701  2 ACCESS_MODE = .IO_PACKET[IRP$V_MODE];
;  390     0702  2
;  391     0703  2 ! Scan the buffer packet, picking up each entry. The first byte of the
;  392     0704  2 ! text is the attribute code, and must be overwritten with the access
;  393     0705  2 ! mode of the request for the I/O completion processing.
;  394     0706  2 !
;  395     0707  2
;  396     0708  2 INCR I FROM ABD$C_ATTRIB TO .IO_PACKET[IRP$W_BCNT]-1 DO                    ;
;  397     0709  3     BEGIN
;  398     0710  3     P = .ABD[.I, ABD$W_TEXT] + ABD[.I, ABD$W_TEXT];
;  399     0711  3     COUNT = .ABD[.I, ABD$W_COUNT];
;  400     0712  3     CODE = .(.P)<0,8> - 1;
;  401     0713  3     (.P)<0,8> = .ACCESS_MODE;
;  402     0714  3     P = .P + 1;
;  403     0715  3
;  404     0716  3 ! Check the attribute code for legality, and then check the requested
;  405     0717  3 ! size against the limit. If an error exit is made, first truncate the
;  406     0718  3 ! descriptor count to inhibit return of the unprocessed descriptors.
;  407     0719  3 !
;  408     0720  3
;  409     0721  3     IF .CODE GTR MAX_CODE - 1
;  410     0722  3     THEN
;  411     0723  4         BEGIN
;  412     0724  4         IO_PACKET[IRP$W_BCNT] = .I;
;  413     0725  4         (ERR_STATUS (SS$_BADATTRIB); RETURN 0);
;  414     0726  3         END;
```

```
;  415     0727  3
;  416     0728  3      MAX_COUNT = .ATC[.CODE, ATC_MAX_SIZE];
;  417     0729  3      IF .COUNT GTR .MAX_COUNT
;  418     0730  3      THEN
;  419     0731  4          BEGIN
;  420     0732  4          IO_PACKET[IRP$W_BCNT] = .I;
;  421     0733  4          (ERR_STATUS (SS$_BADATTRIB); RETURN 0);
;  422     0734  3          END;
;  423     0735  3
;  424     0736  3
;  425     0737  3      ! Get the action routine code first.
;  426     0738  3      !
;  427     0739  3
;  428     0740  3      ACTION = .ATC[.CODE, ATC_ACTION];
;  429     0741  3
;  430     0742  3      ! Compute the address of the attribute.
;  431     0743  3      !
;  432     0744  3
;  433     0745  3      ADDRESS =
;  434     0746  4          (
;  435     0747  4          CASE .ATC[.CODE, ATC_LOCATION] FROM 0 TO ATC_LASTATC OF
;  436     0748  4              SET
;  437     0749  4              [ATC_ZERO]:         ATT_BUFFER;
;  438     0750  4              [ATC_FCB]:          .PRIMARY_FCB;
;  439     0751  4              [ATC_HEADER]:       .LOCAL_HEADER;
;  440     0752  4              [ATC_IDENT]:        .LOCAL_HEADER + .LOCAL_HEADER[FH1$B_IDOFFSET]*2;
;  441     0753  4              [ATC_MAP]:          .LOCAL_HEADER + .LOCAL_HEADER[FH1$B_MPOFFSET]*2;
;  442     0754  4              [ATC_ACL]:          .LOCAL_HEADER;
;  443     0755  5              [ATC_ACPGBL]:       BEGIN
;  444     0756  5                                  ACTION = ACT_ZERO;
;  445     0757  5                                  ATT_BUFFER
;  446     0758  4                                  END;
;  447     0759  5              [ATC_FID2NAME]:     BEGIN
;  448     0760  5                                  FID_TO_SPEC ();
;  449     0761  5                                  LOCAL_HEADER = .FILE_HEADER;
;  450     0762  5                                  FILE_SPEC_LEN
;  451     0763  4                                  END;
;  452     0764  4              TES
;  453     0765  4          )
;  454     0766  3          + .ATC[.CODE, ATC_OFFSET];
;  455     0767  3
;  456     0768  3
;  457     0769  3      ! Finally execute the action routine.
;  458     0770  3      !
;  459     0771  3
;  460     0772  3      CASE .ACTION FROM 0 TO ACT_LASTACT OF
;  461     0773  3          SET
;  462     0774  3
;  463     0775  3          [ACT_NOP]:      COUNT = 0;
;  464     0776  3
;  465     0777  4          [ACT_ILLEGAL]:  BEGIN
;  466     0778  4                          IO_PACKET[IRP$W_BCNT] = .I;
;  467     0779  4                          (ERR_STATUS (SS$_BADATTRIB); RETURN 0);
;  468     0780  3                          END;
;  469     0781  3
;  470     0782  3          [ACT_COPY]:     0;
;  471     0783  3
```

```
;  472     0784  4          [ACT_UCHAR]:    BEGIN
;  473     0785  4                          (ATT_BUFFER)<0,32> = .(.ADDRESS)<0,16>;
;  474     0786  4                          ADDRESS = ATT_BUFFER;
;  475     0787  3                          END;
;  476     0788  3
;  477     0789  4          [ACT_STATBLK]:  BEGIN
;  478     0790  4                          ATT_BUFFER[SBK$L_STLBN] = ROT (.ADDRESS[FCB$L_STLBN], 16);
;  479     0791  4                          ATT_BUFFER[SBK$L_FILESIZE] = ROT (.ADDRESS[FCB$L_FILESIZE], 16);
;  480     0792  4                          ATT_BUFFER[SBK$B_ACNT]  = .ADDRESS[FCB$W_ACNT];
;  481     0793  4                          ATT_BUFFER[SBK$B_LCNT]  = .ADDRESS[FCB$W_LCNT];
;  482     0794  4                          ATT_BUFFER[SBK$L_FCB] = .ADDRESS;
;  483     0795  4                          (ATT_BUFFER[SBK$L_FCB]+4)<0,16> = 0; ! unused field
;  484     0796  4                          ATT_BUFFER[SBK$W_ACNT] = .ADDRESS[FCB$W_ACNT];
;  485     0797  4                          ATT_BUFFER[SBK$W_LCNT] = .ADDRESS[FCB$W_LCNT];
;  486     0798  4                          ATT_BUFFER[SBK$W_WCNT] = .ADDRESS[FCB$W_WCNT];
;  487     0799  4                          ATT_BUFFER[SBK$W_TCNT] = .ADDRESS[FCB$W_TCNT];
;  488     0800  4                          ATT_BUFFER[SBK$L_READS] = 0;
;  489     0801  4                          ATT_BUFFER[SBK$L_WRITES] = 0;
;  490     0802  4                          IF .CURRENT_WINDOW NEQ 0
;  491     0803  4                          THEN
;  492     0804  5                              BEGIN
;  493     0805  5                              ATT_BUFFER[SBK$L_READS] = .CURRENT_WINDOW[WCB$L_READS];
;  494     0806  5                              ATT_BUFFER[SBK$L_WRITES] = .CURRENT_WINDOW[WCB$L_WRITES];
;  495     0807  4                              END;
;  496     0808  4                          ADDRESS = ATT_BUFFER;
;  497     0809  3                          END;
;  498     0810  3
;  499     0811  3          [ACT_BLOCKSIZE]: ADDRESS = UPLIT (512);
;  500     0812  3
;  501     0813  4          [ACT_ZERO]:     BEGIN
;  502     0814  4                          CH$FILL (0, .COUNT, .P);
;  503     0815  4                          COUNT = 0;
;  504     0816  3                          END;
;  505     0817  3
;  506     0818  4          [ACT_BLANK]:    BEGIN
;  507     0819  4                          ADDRESS = UPLIT BYTE (' ');
;  508     0820  3                          END;
;  509     0821  3
;  510     0822  4          [ACT_UIC]:      BEGIN
;  511     0823  4                          MAP ATT_BUFFER : VECTOR [,WORD];
;  512     0824  4                          ATT_BUFFER[0] = .(.ADDRESS)<0,8>;
;  513     0825  4                          ATT_BUFFER[1] = .(.ADDRESS)<8,8>;
;  514     0826  4                          ADDRESS = ATT_BUFFER;
;  515     0827  3                          END;
;  516     0828  3
;  517     0829  4          [ACT_ASCNAME]:  BEGIN
;  518     0830  4                          CH$FILL (' ', 86, ATT_BUFFER);
;  519     0831  4                          MAKE_STRING (.ADDRESS-6, ATT_BUFFER);
;  520     0832  4                          ADDRESS = ATT_BUFFER;
;  521     0833  3                          END;
;  522     0834  3
;  523     0835  4          [ACT_DATIME]:   BEGIN
;  524     0836  4                          (DATBUF+00)<0,16> = .(.ADDRESS);
;  525     0837  4                          (DATBUF+02)<0,8>  = '-';
;  526     0838  4                          (DATBUF+03)<0,24> = .(.ADDRESS+2);
;  527     0839  4                          (DATBUF+06)<0,24> = '-19';
;  528     0840  4                          (DATBUF+09)<0,16> = .(.ADDRESS+5);
```

```
;  529     0841  4                    (DATBUF+11)<0,8>  = ' ';
;  530     0842  4                    IF .(.ADDRESS+7)<0,8> NEQ 0
;  531     0843  4                    THEN
;  532     0844  5                        BEGIN
;  533     0845  5                        (DATBUF+12)<0,16> = .(.ADDRESS+7);
;  534     0846  5                        (DATBUF+14)<0,8>  = ':';
;  535     0847  5                        (DATBUF+15)<0,16> = .(.ADDRESS+9);
;  536     0848  5                        (DATBUF+17)<0,8>  = ':';
;  537     0849  5                        (DATBUF+18)<0,16> = .(.ADDRESS+11);
;  538     0850  5                        END
;  539     0851  4                    ELSE
;  540     0852  5                        BEGIN
;  541     0853  5                        (DATBUF+12)<0,32> = '00:0';
;  542     0854  5                        (DATBUF+16)<0,32> = '0:00';
;  543     0855  4                        END;
;  544     0856  4                    (DATBUF+20)<0,24> = '.00';
;  545     0857  4                    (ATT_BUFFER) = (ATT_BUFFER+4) = 0;
;  546     0858  4                    DATDESC[0] = DATLEN;
;  547     0859  4                    DATDESC[1] = DATBUF;
;  548     0860  4                    SYS$BINTIM (DATDESC, ATT_BUFFER);
;  549     0861  4                    ADDRESS = ATT_BUFFER;
;  550     0862  3                    END;
;  551     0863  3
;  552     0864  4            [ACT_ACL]:      BEGIN
;  553     0865  4                    ACE_POINTER = .P;
;  554     0866  4                    CASE .CODE + 1 FROM ATR$C_ADDACLENT TO ATR$C_READACE OF
;  555     0867  4                    SET
;  556     0868  4                        [ATR$C_ADDACLENT,
;  557     0869  4                         ATR$C_DELACLENT,
;  558     0870  4                         ATR$C_MODACLENT,
;  559     0871  4                         ATR$C_DELETEACL]:  0;
;  560     0872  4
;  561     0873  4                        [ATR$C_FNDACLENT,
;  562     0874  4                         ATR$C_FNDACETYP,
;  563     0875  4                         ATR$C_READACL,
;  564     0876  5                         ATR$C_READACE]:    BEGIN
;  565     0877  5                                            CH$FILL (0, .COUNT, .ACE_POINTER);
;  566     0878  5                                            ACE_POINTER[ACE$W_FLAGS] = SS$_ACLEMPTY;
;  567     0879  4                                            END;
;  568     0880  4
;  569     0881  5                        [ATR$C_ACLLENGTH]: BEGIN
;  570     0882  5                                            CH$FILL (0, .COUNT, .P);
;  571     0883  4                                            END;
;  572     0884  4
;  573     0885  4                    TES;
;  574     0886  4                    COUNT = 0;
;  575     0887  3                    END;
;  576     0888  3
;  577     0889  4            [ACT_ACMODE]:   BEGIN
;  578     0890  4                    (.P-1)<0,8> = .IO_PACKET[IRP$V_MODE];
;  579     0891  4                    ACCESS_MODE = MAXU (.IO_PACKET[IRP$V_MODE], .(.P)<0,8>);
;  580     0892  4                    COUNT = 0;
;  581     0893  3                    END;
;  582     0894  3
;  583     0895  3            TES;
;  584     0896  3            CH$MOVE (.COUNT, .ADDRESS, .P); ! finally copy the attribute
;  585     0897  3
```

```
;  586          0898  3
;  587          0899  3
;  588          0900  2     END;                                   ! end of loop
;  589          0901  2
;  590          0902  2 ! Stop metering of this subfunction
;  591          0903  2 !
;  592          0904  2
;  593          0905  2 PMS_END_SUB ();
;  594          0906  2
;  595          0907  2 RETURN .STATUS;                            ! return success
;  596          0908  2
;  597          0909  1 END;                                       ! end of routine READ_ATTRIB

                                                        .TITLE  RWATTR
                                                        .IDENT  \V04-000\

                                                        .PSECT  $CODE$,NOWRT,2

                                02  08  02  02  00000 P.AAA:  .BYTE   2, 2, 8, 2
                                    0000  0005  00004          .WORD   5, 0
                                02  0A  02  02  00008          .BYTE   2, 2, 10, 2
                                    0000  0003  0000C          .WORD   3, 0
                                09  0C  02  04  00010          .BYTE   4, 2, 12, 9
                                    0000  0004  00014          .WORD   4, 0
                                02  0E  02  04  00018          .BYTE   4, 2, 14, 2
                                    0000  0020  0001C          .WORD   32, 0
                                02  00  03  00  00020          .BYTE   0, 3, 0, 2
                                    0000  000A  00024          .WORD   10, 0
                                02  06  03  00  00028          .BYTE   0, 3, 6, 2
                                    0000  0004  0002C          .WORD   4, 0
                                02  08  03  00  00030          .BYTE   0, 3, 8, 2
                                    0000  0002  00034          .WORD   2, 0
                                02  26  03  02  00038          .BYTE   2, 3, 38, 2
                                    0000  0007  0003C          .WORD   7, 0
                                03  00  01  01  00040          .BYTE   1, 1, 0, 3
                                    0000  0020  00044          .WORD   32, 0
                                02  00  02  01  00048          .BYTE   1, 2, 0, 2
                                    0000  0200  0004C          .WORD   512, 0
                                05  00  00  00  00050          .BYTE   0, 0, 0, 5
                                    0000  0002  00054          .WORD   2, 0
                                04  00  00  00  00058          .BYTE   0, 0, 0, 4
                                    0000  0050  0005C          .WORD   80, 0
                                02  0A  03  02  00060          .BYTE   2, 3, 10, 2
                                    0000  0023  00064          .WORD   35, 0
                                00  00  00  00  00068          .BYTE   0, 0, 0, 0
                                    0000  000E  0006C          .WORD   14, 0
                                00  00  00  00  00070          .BYTE   0, 0, 0, 0
                                    0000  0001  00074          .WORD   1, 0
                                06  00  03  00  00078          .BYTE   0, 3, 0, 6
                                    0000  0056  0007C          .WORD   86, 0
                                07  19  03  02  00080          .BYTE   2, 3, 25, 7
                                    0000  0008  00084          .WORD   8, 0
                                07  0C  03  02  00088          .BYTE   2, 3, 12, 7
                                    0000  0008  0008C          .WORD   8, 0
                                07  26  03  02  00090          .BYTE   2, 3, 38, 7
                                    0000  0008  00094          .WORD   8, 0
```

```
04  00  00  02  00098          .BYTE   2, 0, 0, 4
    0000  0008  0009C          .WORD   8, 0
08  08  02  02  000A0          .BYTE   2, 2, 8, 8
    0000  0004  000A4          .WORD   4, 0
02  0A  02  02  000A8          .BYTE   2, 2, 10, 2
    0000  0002  000AC          .WORD   2, 0
04  00  00  02  000B0          .BYTE   2, 0, 0, 4
    0000  0002  000B4          .WORD   2, 0
04  00  00  02  000B8          .BYTE   2, 0, 0, 4
    0000  0002  000BC          .WORD   2, 0
04  00  00  02  000C0          .BYTE   2, 0, 0, 4
    0000  0008  000C4          .WORD   8, 0
08  08  02  01  000C8          .BYTE   1, 2, 8, 8
    0000  0004  000CC          .WORD   4, 0
04  00  00  01  000D0          .BYTE   1, 0, 0, 4
    0000  0002  000D4          .WORD   2, 0
04  00  00  02  000D8          .BYTE   2, 0, 0, 4
    0000  0006  000DC          .WORD   6, 0
04  00  00  02  000E0          .BYTE   2, 0, 0, 4
    0000  0002  000E4          .WORD   2, 0
0A  00  00  02  000E8          .BYTE   2, 0, 0, 10
    0000  0001  000EC          .WORD   1, 0
0B  00  00  04  000F0          .BYTE   4, 0, 0, 11
    0000  00FF  000F4          .WORD   255, 0
0B  00  05  04  000F8          .BYTE   4, 5, 0, 11
    0000  00FF  000FC          .WORD   255, 0
0B  00  05  04  00100          .BYTE   4, 5, 0, 11
    0000  00FF  00104          .WORD   255, 0
0B  00  05  01  00108          .BYTE   1, 5, 0, 11
    0000  00FF  0010C          .WORD   255, 0
0B  00  05  01  00110          .BYTE   1, 5, 0, 11
    0000  00FF  00114          .WORD   255, 0
0B  00  05  04  00118          .BYTE   4, 5, 0, 11
    0000  00FF  0011C          .WORD   255, 0
0B  00  05  01  00120          .BYTE   1, 5, 0, 11
    0000  0200  00124          .WORD   512, 0
0B  00  05  01  00128          .BYTE   1, 5, 0, 11
    0000  0004  0012C          .WORD   4, 0
0B  00  05  01  00130          .BYTE   1, 5, 0, 11
    0000  00FF  00134          .WORD   255, 0
04  00  00  02  00138          .BYTE   2, 0, 0, 4
    0000  017C  0013C          .WORD   380, 0
04  00  02  01  00140          .BYTE   1, 2, 0, 4
    0000  0004  00144          .WORD   4, 0
01  00  06  01  00148          .BYTE   1, 6, 0, 1
    0000  0004  0014C          .WORD   4, 0
02  00  06  01  00150          .BYTE   1, 6, 0, 2
    0000  0004  00154          .WORD   4, 0
04  01  06  01  00158          .BYTE   1, 6, 1, 4
    0000  00FF  0015C          .WORD   255, 0
0C  00  00  01  00160          .BYTE   1, 0, 0, 12
    0000  0001  00164          .WORD   1, 0
02  00  07  01  00168          .BYTE   1, 7, 0, 2
    0000  0200  0016C          .WORD   512, 0
04  00  00  00  00170          .BYTE   0, 0, 0, 4
    0000  0000  00174          .WORD   0, 0
00  00  00  00  00178          .BYTE   0, 0, 0, 0
```

```
                        0000  0000  0017C           .WORD   0, 0                                                    ;
                          00000200  00180 P.AAB:    .LONG   512                                                     ;
                                20  00184 P.AAC:    .ASCII  \ \                                                     ;

                                                    .PSECT  $LOCKEDD1$,NOEXE,2

                                                00000 FILE_SPEC_LEN:
                                                    .BLKB   2
                                                00002 FULL_FILE_SPEC:
                                                    .BLKB   510

                                                ATC=                P.AAA
                                                    .EXTRN  IO_PACKET, PRIMARY_FCB
                                                    .EXTRN  CURRENT_WINDOW, FILE_HEADER
                                                    .EXTRN  PMS_START_SUB, PMS_END_SUB
                                                    .EXTRN  MAKE_STRING, SYS$BINTIM
                                                    .EXTRN  READ_HEADER, USER_STATUS

                                                    .PSECT  $CODE$,NOWRT,2

                                    0FFC 00000      .ENTRY  READ_ATTRIB, Save R2,R3,R4,R5,R6,R7,R8,R9,-             ; 0613
                                                            R10,R11                                                 ;
                              5E    FF70   CE  9E 00002     MOVAB   -144(SP), SP                                    ;
                                           09  DD 00007     PUSHL   #9                                              ; 0691
                          0000G   CF       01  FB 00009     CALLS   #1, PMS_START_SUB                               ;
                              14  AE       01  D0 0000E     MOVL    #1, STATUS                                      ; 0693
                              56    04     AC  D0 00012     MOVL    HEADER, LOCAL_HEADER                            ; 0694
                              50  0000G    CF  D0 00016     MOVL    IO_PACKET, R0                                   ; 0700
                              2A  A0       02  88 0001B     BISB2   #2, 42(R0)                                      ;
    10  AE        0B  A0                 02  00  EF 0001F     EXTZV   #0, #2, 11(R0), ACCESS_MODE                     ; 0701
                              0C  AE       32  A0  3C 00026     MOVZWL  50(R0), 12(SP)                          ; 0708
                              5A           04  D0  0002B     MOVL    #4, I                                           ; 0710
                                       021B    31  0002E     BRW     37$                                             ;
                              50         08 BC4A   7E 00031 1$:  MOVAQ   @ABD[I], R0                                ;
                              59             60    3C 00036     MOVZWL  (R0), P                                 ;
                              59             50    C0 00039     ADDL2   R0, P                                   ;
                              5B           02  A0  3C 0003C     MOVZWL  2(R0), COUNT                            ; 0711
                              58               69  9A 00040     MOVZBL  (P), CODE                               ; 0712
                                               58  D7 00043     DECL    CODE                                    ;
                              89           10  AE  90 00045     MOVB    ACCESS_MODE, (P)+                       ; 0713
                              2F               58  D1 00049     CMPL    CODE, #47                               ; 0721
                                               0B  14 0004C     BGTR    2$                                      ;
                                     FE2C CF48   7F 0004E     PUSHAQ  ATC+4[CODE]                             ; 0728
                              6E               9E  3C 00053     MOVZWL  @(SP)+, MAX_COUNT                       ;
                              6E               5B  D1 00056     CMPL    COUNT, MAX_COUNT                        ; 0729
                                               03  15 00059 2$:  BLEQ    3$                                      ;
                                             0085  31 0005B     BRW     15$                                     ;
                                     FE1B CF48   7F 0005E 3$:  PUSHAQ  ATC+3[CODE]                             ; 0740
                                  04  AE       9E  90 00063     MOVB    @(SP)+, ACTION                          ;
                                     FE10 CF48   7F 00067     PUSHAQ  ATC+1[CODE]                             ; 0747
                                  07           00  9E  8F 0006C     CASEB   @(SP)+, #0, #7                          ;
                        0019      0028       0012        0031       00070 4$:  .WORD   11$-4$,-                        ;
                        0037      002D       0028        001E       00078              5$-4$,-                         ;
                                                                                      9$-4$,-                         ;
                                                                                      6$-4$,-                         ;
                                                                                      7$-4$,-                         ;
                                                                                      9$-4$,-                         ;
```

```
                                                                                                                                10$-4$,-                                                                                                                                ;
                                                                                                                                12$-4$                                                                                                                                  ;
                                                                                                                                                                                                                                                                        ;
                                                                                                                                                                                                                                                                        ;
                                                                                                                                                                                                                                                                        
```

```
                                                 2E  11 00143           BRB     24$                                             ; 0808
                                  57      FEB2   CF  9E 00145 19$:      MOVAB   P.AAB, ADDRESS                                  ; 0811
                                                 05  11 0014A           BRB     21$                                             ;
                                  57      FEAF   CF  9E 0014C 20$:      MOVAB   P.AAC, ADDRESS                                  ; 0819
                                         00F4    31  00151 21$:      BRW     36$                                             ; 0772
                            18    AE            67  9B 00154 22$:      MOVZBW  (ADDRESS), ATT_BUFFER                           ; 0824
                            1A    AE      01    A7  9B 00158           MOVZBW  1(ADDRESS), ATT_BUFFER+2                        ; 0825
                                                 14  11 0015D           BRB     24$                                             ; 0826
    0056    8F          20                6E        00  2C 0015F 23$:      MOVC5   #0, (SP), #32, #86, ATT_BUFFER                  ; 0830
                                                 18  AE  00166                                                                   ;
                                                 18  AE  9F 00168           PUSHAB  ATT_BUFFER                                      ; 0831
                                                 FA  A7  9F 0016B           PUSHAB  -6(ADDRESS)                                     ;
                                0000G    CF               02  FB 0016E           CALLS   #2, MAKE_STRING                                 ;
                                                 73  11 00173 24$:      BRB     28$                                             ; 0832
                            78    AE            67  B0 00175 25$:      MOVW    (ADDRESS), DATBUF                               ; 0836
                            7A    AE            2D  90 00179           MOVB    #45, DATBUF+2                                   ; 0837
    7B      AE          18                00      02  A7  F0 0017D           INSV    2(ADDRESS), #0, #24, DATBUF+3                   ; 0838
    7E      AE          18                00 0039312D   8F  F0 00184           INSV    #3748141, #0, #24, DATBUF+6                     ; 0839
                            F1    AD            05  A7  B0 0018E           MOVW    5(ADDRESS), DATBUF+9                            ; 0840
                            F3    AD            20  90 00193           MOVB    #32, DATBUF+11                                  ; 0841
                                                 07  A7  95 00197           TSTB    7(ADDRESS)                                      ; 0842
                                                 19  13 0019A           BEQL    26$                                             ;
                            F4    AD            07  A7  B0 0019C           MOVW    7(ADDRESS), DATBUF+12                           ; 0845
                            F6    AD            3A  90 001A1           MOVB    #58, DATBUF+14                                  ; 0846
                            F7    AD            09  A7  B0 001A5           MOVW    9(ADDRESS), DATBUF+15                           ; 0847
                            F9    AD            3A  90 001AA           MOVB    #58, DATBUF+17                                  ; 0848
                            FA    AD            0B  A7  B0 001AE           MOVW    11(ADDRESS), DATBUF+18                          ; 0849
                                                 10  11 001B3           BRB     27$                                             ; 0842
                            F4    AD 303A3030   8F  D0 001B5 26$:      MOVL    #809119792, DATBUF+12                           ; 0853
                            F8    AD 30303A30   8F  D0 001BD           MOVL    #808466992, DATBUF+16                           ; 0854
    FC      AD          18                00 0030302E   8F  F0 001C5 27$:      INSV    #3158062, #0, #24, DATBUF+20                    ; 0856
                                                 18  AE  7C 001CF           CLRQ    ATT_BUFFER                                      ; 0857
                            70    AE            17  D0 001D2           MOVL    #23, DATDESC                                    ; 0858
                            74    AE      78    AE  9E 001D6           MOVAB   DATBUF, DATDESC+4                               ; 0859
                                                 18  AE  9F 001DB           PUSHAB  ATT_BUFFER                                      ; 0860
                                                 74  AE  9F 001DE           PUSHAB  DATDESC                                         ;
                                00000000G    9F               02  FB 001E1           CALLS   #2, @#SYS$BINTIM                                ;
                                             57      18  AE  9E 001E8 28$:      MOVAB   ATT_BUFFER, ADDRESS                             ; 0861
                                                 5A  11 001EC           BRB     36$                                             ; 0772
                                08    AE            59  D0 001EE 29$:      MOVL    P, ACE_POINTER                                  ; 0865
                          08                      1E  58  CF 001F2           CASEL   CODE, #30, #8                                   ; 0866
    0014          0050                0050          0050      001F6 30$:      .WORD   35$-30$,-                                       ;
    0027          0014                0050          0014      001FE                   35$-30$,-                                       ;
                                                    0014      00206                   35$-30$,-                                       ;
                                                                                      31$-30$,-                                       ;
                                                                                      31$-30$,-                                       ;
                                                                                      35$-30$,-                                       ;
                                                                                      31$-30$,-                                       ;
                                                                                      32$-30$,-                                       ;
                                                                                      31$-30$                                         ;
                                                 3C  11 00208           BRB     35$                                             ;
    5B          00                6E                00  2C 0020A 31$:      MOVC5   #0, (SP), #0, COUNT, @ACE_POINTER               ; 0877
                                                 08  BE  0020F                                                                   ;
                            50    08    AE            02  C1 00211           ADDL3   #2, ACE_POINTER, R0                             ; 0878
                                             60      09D0    8F  B0 00216           MOVW    #2512, (R0)                             ;
                                                 29  11 0021B           BRB     35$                                             ; 0866
    5B          00                6E                00  2C 0021D 32$:      MOVC5   #0, (SP), #0, COUNT, (P)                        ; 0882
```

```
                                                 69     00222
                                                 21  11 00223          BRB     35$                                                 ;
                                          50     CF  D0 00225 33$:      MOVL    IO_PACKET, R0                                       ; 0886
                                          02     00  EF 0022A           EXTZV   #0, #2, 11(R0), R1                                  ; 0890
                                       FF 51     51  90 00230          MOVB    R1, -1(P)                                           ;
                                          02     00  EF 00234           EXTZV   #0, #2, 11(R0), R0                                  ;
                                          50     69  91 0023A           CMPB    (P), R0                                             ; 0891
                                                 03  1B 0023D           BLEQU   34$                                                 ;
                                          50     69  9A 0023F           MOVZBL  (P), R0                                             ;
                                       10 AE     50  D0 00242 34$:      MOVL    R0, ACCESS_MODE                                     ;
                                                 5B  D4 00246 35$:      CLRL    COUNT                                               ; 0892
                                    69            5B  28 00248 36$:      MOVC3   COUNT, (ADDRESS), (P)                               ; 0896
                                    02     5A  0C AE  F2 0024C 37$:      AOBLSS  12(SP), I, 38$                                      ; 0708
                                                 03  11 00251           BRB     39$                                                 ;
                                              FDDB  31 00253 38$:       BRW     1$                                                  ;
                                 0000G CF        00  FB 00256 39$:      CALLS   #0, PMS_END_SUB                                     ; 0905
                                          50  14 AE  D0 0025B           MOVL    STATUS, R0                                          ; 0907
                                                 04  0025F             RET                                                         ;
                                                 50  D4 00260 40$:      CLRL    R0                                                  ; 0909
                                                     04 00262           RET                                                         ;

; Routine Size:  611 bytes,    Routine Base:  $CODE$ + 0185
```

```
;  599     0910  1 ROUTINE FID_TO_SPEC : NOVALUE =
;  600     0911  1
;  601     0912  1 !++
;  602     0913  1 !
;  603     0914  1 ! FUNCTIONAL DESCRIPTION:
;  604     0915  1 !
;  605     0916  1 !       This routine converts the specified file-ID (contained in the header
;  606     0917  1 !       supplied) to a full file specification.  If the directory-ID in
;  607     0918  1 !       the FIB is non-zero, the directory portion of the spec is filled
;  608     0919  1 !       in.  Otherwise, it is left blank (e.g., []).
;  609     0920  1 !
;  610     0921  1 ! CALLING SEQUENCE:
;  611     0922  1 !       FID_TO_SPEC ()
;  612     0923  1 !
;  613     0924  1 ! INPUT PARAMETERS:
;  614     0925  1 !       none
;  615     0926  1 !
;  616     0927  1 ! IMPLICIT INPUTS:
;  617     0928  1 !       none
;  618     0929  1 !
;  619     0930  1 ! OUTPUT PARAMETERS:
;  620     0931  1 !       none
;  621     0932  1 !
;  622     0933  1 ! IMPLICIT OUTPUTS:
;  623     0934  1 !       none
;  624     0935  1 !
;  625     0936  1 ! ROUTINE VALUE:
;  626     0937  1 !       none
;  627     0938  1 !
;  628     0939  1 ! SIDE EFFECTS:
;  629     0940  1 !       none
;  630     0941  1 !
;  631     0942  1 !--
;  632     0943  1
;  633     0944  2 BEGIN
;  634     0945  2
;  635     0946  2 LINKAGE
;  636     0947  2         L_CVT_DEVNAM    = JSB (REGISTER = 0,    ! Buffer length
;  637     0948  2                                REGISTER = 1,    ! Buffer address
;  638     0949  2                                REGISTER = 4,    ! Cluster node conversion flag
;  639     0950  2                                REGISTER = 5;    ! UCB address
;  640     0951  2                                REGISTER = 1);   ! Length of converted name
;  641     0952  2
;  642     0953  2 LITERAL
;  643     0954  2         NODE_LEN        = 15,                   ! Maximum cluster node name length
;  644     0955  2         DEVNAME_LEN     = 15,                   ! Maximum device name length
;  645     0956  2         UNIT_LEN        = 5,                    ! Maximum unit number length
;  646     0957  2         FILENAME_LEN    = 9,                    ! Maximun file name length
;  647     0958  2         FILETYPE_LEN    = 3,                    ! Maximum file type length
;  648     0959  2         FILEVER_LEN     = 5,                    ! Maximum version length
;  649     0960  2
;  650     0961  2         FULLDEV_LEN     = NODE_LEN + 1 +        ! Full device spec length
;  651     0962  2                           DEVNAME_LEN + UNIT_LEN + 1,
;  652     0963  2         FULLDIR_LEN     = 1 + FILENAME_LEN + 1, ! Max dir spec length
;  653     0964  2         FULLFILE_LEN    = FILENAME_LEN + 1 +    ! Max file name length
;  654     0965  2                           FILETYPE_LEN + 1 +    !   including type & version
;  655     0966  2                           FILEVER_LEN,
```

```
;  656     0967  2          FULLSPEC_LEN    = 2 + FULLDEV_LEN +     ! Maximum file spec length
;  657     0968  2                            FULLDIR_LEN +         ! Includes word size prefix
;  658     0969  2                            FULLFILE_LEN;
;  659     0970  2
;  660     0971  2 LOCAL
;  661     0972  2          DEVICE_LEN,                              ! Length of the device name
;  662     0973  2          DIR_HEADER      : REF BBLOCK,            ! Current directory file header
;  663     0974  2          IDENT_AREA      : REF BBLOCK;            ! Address of header ident area
;  664     0975  2
;  665     0976  2 EXTERNAL
;  666     0977  2          FILE_HEADER     : REF BBLOCK,            ! Address of current file header
;  667     0978  2          IO_PACKET       : REF BBLOCK,            ! Address of current I/O packet
;  668     0979  2          LOCAL_FIB       : BBLOCK,                ! Copy of user's FIB
;  669     0980  2          PRIMARY_FCB     : REF BBLOCK;            ! Address of primary FCB
;  670     0981  2
;  671     0982  2 EXTERNAL ROUTINE
;  672     0983  2          READ_HEADER,                             ! Read & validate file header
;  673     0984  2          MAKE_STRING,                             ! Convert file name block to string
;  674     0985  2          IOC$CVT_DEVNAM  : L_CVT_DEVNAM ADDRESSING_MODE (GENERAL);
;  675     0986  2
;  676     0987  2 BIND
;  677     0988  2          DEVICE_NAME     = FULL_FILE_SPEC         : VECTOR [,BYTE],       ! Device name spec
;  678     0989  2          DIR_NAME        = FULL_FILE_SPEC[FULLDEV_LEN + 1]       : VECTOR [,BYTE],       ! Directory name sto
;  679     0990  2          FILE_NAME       = FULL_FILE_SPEC[FULLDEV_LEN + 1 +      ! File name storage
;  680     0991  2                                          FULLDIR_LEN]    : VECTOR [,BYTE];
;  681     0992  2
;  682     0993  2 ! Initialize all of the necessary storage.
;  683     0994  2
;  684     0995  2 CH$FILL (0, FULLSPEC_LEN, FULL_FILE_SPEC);
;  685     0996  2
;  686     0997  2 ! Get the device name.
;  687     0998  2
;  688     0999  2 IOC$CVT_DEVNAM (FULLDEV_LEN, DEVICE_NAME, 0, .IO_PACKET[IRP$L_UCB], DEVICE_LEN);
;  689     1000  2 FILE_SPEC_LEN[0] = .DEVICE_LEN;
;  690     1001  2
;  691     1002  2 ! Start building the directory specification.
;  692     1003  2
;  693     1004  2 FULL_FILE_SPEC[.FILE_SPEC_LEN[0]] = '[';
;  694     1005  2 FILE_SPEC_LEN[0] = .FILE_SPEC_LEN[0] + 1;
;  695     1006  2
;  696     1007  2 ! If a directory-ID has been given in the FIB, return the diretory spec
;  697     1008  2 ! also.  Otherwise, leave it blank.
;  698     1009  2
;  699     1010  2 IF .LOCAL_FIB[FIB$W_DID_NUM] NEQ 0
;  700     1011  2 THEN
;  701     1012  3     BEGIN
;  702     1013  3     DIR_HEADER = READ_HEADER (LOCAL_FIB[FIB$W_DID], 0);
;  703     1014  3     IDENT_AREA = .DIR_HEADER + .DIR_HEADER[FHT$B_IDOFFSET]*2;
;  704     1015  3     MAKE_STRING (IDENT_AREA[FI1$W_FILENAME] - $BYTEOFFSET (NMB$W_NAME),
;  705     1016  3                  FULL_FILE_SPEC[.FILE_SPEC_LEN[0]]);
;  706     1017  3     INCR K FROM 0 TO FILENAME_LEN - 1
;  707     1018  3     DO
;  708     1019  4         BEGIN
;  709     1020  4         IF .FULL_FILE_SPEC[.FILE_SPEC_LEN[0]] EQL '.' THEN EXITLOOP;
;  710     1021  4         IF .K EQL FILENAME_LEN - 1
;  711     1022  4         THEN
;  712     1023  5             BEGIN
```

```
;  713          1024  5              FULL_FILE_SPEC[.FILE_SPEC_LEN[0]] = '?';
;  714          1025  5              EXITLOOP;
;  715          1026  4              END;
;  716          1027  4          FILE_SPEC_LEN[0]= .FILE_SPEC_LEN[0] + 1;
;  717          1028  3          END;
;  718          1029  3      READ_HEADER (LOCAL_FIB[FIB$W_FID], .PRIMARY_FCB);
;  719          1030  2      END;
;  720          1031  2
;  721          1032  2  ! Tie off the directory specification.
;  722          1033  2
;  723          1034  2  FULL_FILE_SPEC[.FILE_SPEC_LEN[0]] = ']';
;  724          1035  2  FILE_SPEC_LEN[0] = .FILE_SPEC_LEN[0] + 1;
;  725          1036  2
;  726          1037  2  ! Now add in the file name.
;  727          1038  2
;  728          1039  2  IDENT_AREA = .FILE_HEADER + .FILE_HEADER[FH1$B_IDOFFSET]*2;
;  729          1040  2  MAKE_STRING (IDENT_AREA[FI1$W_FILENAME] - $BYTEOFFSET (NMB$W_NAME),
;  730          1041  2                FULL_FILE_SPEC[.FILE_SPEC_LEN[0]]);
;  731          1042  2  INCR J FROM 0 TO FULLFILE_LEN
;  732          1043  2  DO
;  733          1044  3      BEGIN
;  734          1045  3      IF .FULL_FILE_SPEC[.FILE_SPEC_LEN[0]] EQL ' '
;  735          1046  3      OR .FULL_FILE_SPEC[.FILE_SPEC_LEN[0]] EQL 0 THEN EXITLOOP;
;  736          1047  3      FILE_SPEC_LEN[0] = .FILE_SPEC_LEN[0] + 1;
;  737          1048  2      END;
;  738          1049  2
;  739          1050  2  RETURN;
;  740          1051  2
;  741          1052  1  END;                                   ! End of routine FID_TO_SPEC


                                                DEVICE_NAME=        FULL_FILE_SPEC
                                                DIR_NAME=           FULL_FILE_SPEC+38
                                                FILE_NAME=          FULL_FILE_SPEC+49
                                                        .EXTRN  LOCAL_FIB, IOC$CVT_DEVNAM

                                        0FFC 00000 FID_TO_SPEC:
                                                        .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11        ; 0910
                                    56      0000'   CF  9E 00002        MOVAB   FILE_SPEC_LEN, R6
    0045    8F          00          6E          00  2C 00007        MOVC5   #0, (SP), #0, #69, FULL_FILE_SPEC   ; 0995
                                        02      A6      0000E
                                    52      0000G   CF  D0 00010        MOVL    IO_PACKET, R2                   ; 0999
                                    51        02    A6  9E 00015        MOVAB   DEVICE_NAME, R1
                                    55        1C    A2  D0 00019        MOVL    28(R2), R5
                                                    54  D4 0001D        CLRL    R4
                                    50              25  D0 0001F        MOVL    #37, R0
                                        00000000G   00  16 00022        JSB     IOC$CVT_DEVNAM
                                    66              51  B0 00028        MOVW    DEVICE_LEN, FILE_SPEC_LEN       ; 1000
                                    50              66  3C 0002B        MOVZWL  FILE_SPEC_LEN, R0               ; 1004
                                02 A640             5B  8F 90 0002E        MOVB    #91, FULL_FILE_SPEC[R0]
                                                    66  B6 00034        INCW    FILE_SPEC_LEN                   ; 1005
                                            0000G   CF  B5 00036        TSTW    LOCAL_FIB+10                    ; 1010
                                                    4C  13 0003A        BEQL    4$
                                                    7E  D4 0003C        CLRL    -(SP)                           ; 1013
                                            0000G   CF  9F 0003E        PUSHAB  LOCAL_FIB+10
                            0000G   CF              02  FB 00042        CALLS   #2, READ_HEADER                 ;
```

```
                51              60  9A 00047          MOVZBL  (DIR_HEADER), R1                        ; 1014
                52            6041  3E 0004A          MOVAW   (DIR_HEADER)[R1], IDENT_AREA
                50              66  3C 0004E          MOVZWL  FILE_SPEC_LEN, R0                       ; 1016
                             02 A640  9F 00051          PUSHAB  FULL_FILE_SPEC[R0]
                             FA   A2  9F 00055          PUSHAB  -6(IDENT_AREA)                          ; 1015
            0000G   CF              02  FB 00058          CALLS   #2, MAKE_STRING                         ; 1016
                                    51  D4 0005D          CLRL    K                                       ; 1017
                50              66  3C 0005F 1$:        MOVZWL  FILE_SPEC_LEN, R0                       ; 1020
                2E           02 A640  91 00062          CMPB    FULL_FILE_SPEC[R0], #46
                                 12  13 00067          BEQL    3$
                08               51  D1 00069          CMPL    K, #8                                   ; 1021
                                 07  12 0006C          BNEQ    2$
              02 A640            3F  90 0006E          MOVB    #63, FULL_FILE_SPEC[R0]                 ; 1024
                                 06  11 00073          BRB     3$                                      ; 1023
                                 66  B6 00075 2$:       INCW    FILE_SPEC_LEN                           ; 1027
        E4              51       08  F3 00077          AOBLEQ  #8, R, 1$                               ; 1017
                     0000G   CF  DD  0007B 3$:         PUSHL   PRIMARY_FCB                             ; 1029
                     0000G   CF  9F  0007F             PUSHAB  LOCAL_FIB+4
            0000G   CF              02  FB 00083          CALLS   #2, READ_HEADER
                50              66  3C 00088 4$:        MOVZWL  FILE_SPEC_LEN, R0                       ; 1034
              02 A640         5D  8F  90 0008B          MOVB    #93, FULL_FILE_SPEC[R0]
                                 66  B6 00091          INCW    FILE_SPEC_LEN                           ; 1035
                50      0000G   DF  9A 00093          MOVZBL  @FILE_HEADER, R0                        ; 1039
                52      0000GDF40   3E 00098          MOVAW   @FILE_HEADER[R0], IDENT_AREA
                50              66  3C 0009E          MOVZWL  FILE_SPEC_LEN, R0                       ; 1041
                             02 A640  9F 000A1          PUSHAB  FULL_FILE_SPEC[R0]
                             FA   A2  9F 000A5          PUSHAB  -6(IDENT_AREA)                          ; 1040
            0000G   CF              02  FB 000A8          CALLS   #2, MAKE_STRING                         ; 1041
                                    51  D4 000AD          CLRL    J                                       ; 1042
                50              66  3C 000AF 5$:        MOVZWL  FILE_SPEC_LEN, R0                       ; 1045
                20           02 A640  91 000B2          CMPB    FULL_FILE_SPEC[R0], #32
                                 0C  13 000B7          BEQL    6$
                             02 A640  95 000B9          TSTB    FULL_FILE_SPEC[R0]                      ; 1046
                                 06  13 000BD          BEQL    6$
                                 66  B6 000BF          INCW    FILE_SPEC_LEN                           ; 1047
                        EA       51  F3 000C1          AOBLEQ  #19, J, 5$                              ; 1042
                                     04 000C5 6$:       RET                                             ; 1052

; Routine Size:  198 bytes,    Routine Base:  $CODE$ + 03E8
```

```
;  743          1053  1 GLOBAL ROUTINE WRITE_ATTRIB (HEADER, ABD, CONTROL_ACCESS) : NOVALUE =
;  744          1054  1
;  745          1055  1 !++
;  746          1056  1 !
;  747          1057  1 ! FUNCTIONAL DESCRIPTION:
;  748          1058  1 !
;  749          1059  1 !       This routine performs the write attributes function. The
;  750          1060  1 !       requested attributes are taken from the buffer packet.
;  751          1061  1 !
;  752          1062  1 ! CALLING SEQUENCE:
;  753          1063  1 !       WRITE_ATTRIB (ARG1, ARG2, ARG3)
;  754          1064  1 !
;  755          1065  1 ! INPUT PARAMETERS:
;  756          1066  1 !       ARG1: address of file header
;  757          1067  1 !       ARG2: address of buffer descriptors
;  758          1068  1 !       ARG3: 1 = check for control access to the file
;  759          1069  1 !             0 = no control access check
;  760          1070  1 !
;  761          1071  1 ! IMPLICIT INPUTS:
;  762          1072  1 !       IO_PACKET: I/O packet for this operation
;  763          1073  1 !       PRIMARY_FCB: FCB of file
;  764          1074  1 !
;  765          1075  1 ! OUTPUT PARAMETERS:
;  766          1076  1 !       NONE
;  767          1077  1 !
;  768          1078  1 ! IMPLICIT OUTPUTS:
;  769          1079  1 !       NONE
;  770          1080  1 !
;  771          1081  1 ! ROUTINE VALUE:
;  772          1082  1 !       NONE
;  773          1083  1 !
;  774          1084  1 ! SIDE EFFECTS:
;  775          1085  1 !       attribute data written into appropriate places
;  776          1086  1 !
;  777          1087  1 !--
;  778          1088  1
;  779          1089  2 BEGIN
;  780          1090  2
;  781          1091  2 MAP
;  782          1092  2         HEADER          : REF BBLOCK,   ! file header arg
;  783          1093  2         ABD             : REF BBLOCKVECTOR [,ABD$C_LENGTH];
;  784          1094  2                                         ! buffer descriptor arg
;  785          1095  2
;  786          1096  2 LOCAL
;  787          1097  2         CTL_ACC_GRANTED,                ! Flag indicating control access granted
;  788          1098  2         SAVE_HIBLK,                     ! saved copy of file's HIBLK
;  789          1099  2         SAVE_CHAR,                      ! initial state of protected attributes
;  790          1100  2         ACE_POINTER     : REF BBLOCK;   ! address of user ACE
;  791          1101  2
;  792          1102  2 EXTERNAL
;  793          1103  2         USER_STATUS     : VECTOR,       ! user request status
;  794          1104  2         CLEANUP_FLAGS   : BITVECTOR,    ! cleanup action flags
;  795          1105  2         IO_PACKET       : REF BBLOCK,   ! I/O packet in process
;  796          1106  2         CURRENT_WINDOW  : REF BBLOCK,   ! window of open file
;  797          1107  2         LOCAL_FIB       : BBLOCK,       ! Copy of user's FIB
;  798          1108  2         PRIMARY_FCB     : REF BBLOCK;   ! FCB of file
;  799          1109  2
```

```
;  800     1110  2 EXTERNAL ROUTINE
;  801     1111  2         PMS_START_SUB,                  ! start subfunction metering
;  802     1112  2         PMS_END_SUB,                    ! end subfunction metering
;  803     1113  2         CHECK_PROTECT,                  ! check file protection
;  804     1114  2         CHECKSUM,                       ! checksum a file header
;  805     1115  2         MARK_DIRTY,                     ! mark buffer for write-back
;  806     1116  2         MAKE_NAMEBLOCK,                 ! convert file string to RAD-50 name block
;  807     1117  2         GET_TIME;                       ! get date/time string
;  808     1118  2
;  809     1119  2
;  810     1120  2 ! Start metering for this subfunction.
;  811     1121  2 !
;  812     1122  2
;  813     1123  2 PMS_START_SUB (PMS_RWATT);
;  814     1124  2
;  815     1125  2 CTL_ACC_GRANTED = 0;
;  816     1126  2
;  817     1127  2 ! Set the appropriate cleanup flags and save the initial state of the
;  818     1128  2 ! protected file characteristics.
;  819     1129  2 !
;  820     1130  2
;  821     1131  2 CHECKSUM (.HEADER);
;  822     1132  2 MARK_DIRTY (.HEADER);
;  823     1133  2 CLEANUP_FLAGS[CLF_FIXFCB] = 1;
;  824     1134  2
;  825     1135  2 SAVE_CHAR = .HEADER[FH1$W_FILECHAR] AND PROTECTED_CHAR;
;  826     1136  2 SAVE_HIBLK = .BBLOCK [HEADER[FH1$W_RECATTR], FAT$L_HIBLK];
;  827     1137  2
;  828     1138  2
;  829     1139  2 ! Scan the buffer packet, picking up each entry. The first byte of the
;  830     1140  2 ! text is the attribute code.
;  831     1141  2 !
;  832     1142  2
;  833     1143  2 INCR I FROM ABD$C_ATTRIB TO .IO_PACKET[IRP$W_BCNT]-1 DO
;  834     1144  3     BEGIN
;  835     1145  3
;  836     1146  3     LOCAL
;  837     1147  3         ACTION          : BYTE,         ! action routine code
;  838     1148  3         P,                              ! pointer to attribute text
;  839     1149  3         COUNT,                          ! attribute size desired
;  840     1150  3         ADDRESS         : REF BBLOCK,   ! address of attribute
;  841     1151  3         CODE,                           ! attribute code
;  842     1152  3         MAX_COUNT,                      ! max size of attribute
;  843     1153  3         ATT_BUFFER      : BBLOCK [20],  ! attribute copy buffer
;  844     1154  3         DATBUF          : VECTOR [13, BYTE]; ! buffer to build date string
;  845     1155  3
;  846     1156  3     P = .ABD[.I, ABD$W_TEXT] + ABD[.I, ABD$W_TEXT];
;  847     1157  3     COUNT = .ABD[.I, ABD$W_COUNT];
;  848     1158  3     CODE = .(.P)<0,8> - 1;
;  849     1159  3     P = .P + 1;
;  850     1160  3
;  851     1161  3 ! Check the attribute code for legality, and then check the requested
;  852     1162  3 ! size against the limit.
;  853     1163  3 !
;  854     1164  3
;  855     1165  3     IF .CODE GTR MAX_CODE - 1
;  856     1166  3         THEN ERR_EXIT (SS$_BADATTRIB);
```

```
;  857          1167  3
;  858          1168  3      MAX_COUNT = .ATC[.CODE, ATC_MAX_SIZE];
;  859          1169  3      IF .COUNT GTR .MAX_COUNT
;  860          1170  3          THEN ERR_EXIT (SS$_BADATTRIB);
;  861          1171  3
;  862          1172  3      IF .ATC[.CODE, ATC_PROTECTED]
;  863          1173  3      THEN
;  864          1174  4          BEGIN
;  865          1175  4          IF NOT .CTL_ACC_GRANTED
;  866          1176  4          THEN
;  867          1177  5              BEGIN
;  868          1178  5              BBLOCK [HEADER[FH1$W_RECATTR], FAT$L_HIBLK] = .SAVE_HIBLK;
;  869          1179  5              CHECK_PROTECT (WRATT_ACCESS, .HEADER, .PRIMARY_FCB);
;  870          1180  5              CTL_ACC_GRANTED = 1;
;  871          1181  5              END;
;  872          1182  4          END;
;  873          1183  3
;  874          1184  3  ! Compute the action routine code.
;  875          1185  3  !
;  876          1186  3
;  877          1187  3      ACTION = .ATC[.CODE, ATC_ACTION];
;  878          1188  3      IF .ATC[.CODE, ATC_READ_ONLY]
;  879          1189  3      THEN ACTION = ACT_NOP;
;  880          1190  3
;  881          1191  3  ! Compute the address of the attribute.
;  882          1192  3  !
;  883          1193  3
;  884          1194  3      ADDRESS =
;  885          1195  4          (
;  886          1196  4          CASE .ATC[.CODE, ATC_LOCATION] FROM 0 TO ATC_LASTATC OF
;  887          1197  4              SET
;  888          1198  4              [ATC_ZERO,
;  889          1199  4               ATC_ACL,
;  890          1200  4               ATC_ACPGBL,
;  891          1201  4               ATC_FID2NAME]:     ATT_BUFFER;
;  892          1202  4              [ATC_FCB]:          .PRIMARY_FCB;
;  893          1203  4              [ATC_HEADER]:       .HEADER;
;  894          1204  4              [ATC_IDENT]:        .HEADER + .HEADER[FH1$B_IDOFFSET]*2;
;  895          1205  4              [ATC_MAP]:          .HEADER + .HEADER[FH1$B_MPOFFSET]*2;
;  896          1206  4              TES
;  897          1207  4          )
;  898          1208  3          + .ATC[.CODE, ATC_OFFSET];
;  899          1209  3
;  900          1210  3
;  901          1211  3  ! Finally execute the action routine.
;  902          1212  3  !
;  903          1213  3
;  904          1214  3      CASE .ACTION FROM 0 TO ACT_LASTACT OF
;  905          1215  3          SET
;  906          1216  3
;  907          1217  3          [ACT_NOP,
;  908          1218  3           ACT_BLOCKSIZE,
;  909          1219  3           ACT_ZERO,
;  910          1220  3           ACT_BLANK,
;  911          1221  3           ACT_ACMODE,
;  912          1222  3           ACT_STATBLK]:  0;
;  913          1223  3
```

```
;  914          1224  3          [ACT_ILLEGAL]:  ERR_EXIT (SS$_BADATTRIB);
;  915          1225  3
;  916          1226  4          [ACT_UIC]:      BEGIN
;  917          1227  4                          IF .COUNT GEQU 4
;  918          1228  4                          THEN
;  919          1229  5                              BEGIN
;  920          1230  5                              (.ADDRESS)<0,8> = ..P;
;  921          1231  5                              (.ADDRESS)<8,8> = .(.P+2);
;  922          1232  5                              IF .(.P+1)<0,8> NEQ 0
;  923          1233  5                              OR .(.P+3)<0,8> NEQ 0
;  924          1234  5                              THEN (.ADDRESS)<0,16> = -1;
;  925          1235  4                              END;
;  926          1236  3                          END;
;  927          1237  3
;  928          1238  4          [ACT_UCHAR]:    BEGIN
;  929          1239  4                          (.ADDRESS)<0,8> = ..P;
;  930          1240  4                          IF .COUNT GEQ 2
;  931          1241  4                          THEN (.ADDRESS)<8,8> = .(.P+1);
;  932          1242  3                          END;
;  933          1243  3
;  934          1244  4          [ACT_ASCNAME]:  BEGIN
;  935          1245  4                          MAKE_NAMEBLOCK (LOCAL_FIB, .COUNT, .P, ATT_BUFFER);
;  936          1246  4                          CH$MOVE (10, ATT_BUFFER[NMB$W_NAME], .ADDRESS);
;  937          1247  3                          END;
;  938          1248  3
;  939          1249  4          [ACT_DATIME]:   BEGIN
;  940          1250  4                          CH$COPY (.COUNT, .P, 0, 8, ATT_BUFFER);
;  941          1251  4                          GET_TIME (DATBUF, ATT_BUFFER);
;  942          1252  4                          COUNT =
;  943          1253  5                              (CASE .CODE FROM ATR$C_CREDATE-1 TO ATR$C_BAKDATE-1 OF
;  944          1254  5                              SET
;  945          1255  5                              [ATR$C_CREDATE-1]:  13;
;  946          1256  5                              [ATR$C_REVDATE-1]:  13;
;  947          1257  5                              [ATR$C_EXPDATE-1]:  7;
;  948          1258  5                              [ATR$C_BAKDATE-1]:  0;
;  949          1259  4                              TES);
;  950          1260  4                          CH$MOVE (.COUNT, DATBUF, .ADDRESS);
;  951          1261  3                          END;
;  952          1262  3
;  953          1263  3          [ACT_COPY]:     CH$MOVE (.COUNT, .P, .ADDRESS);
;  954          1264  3
;  955          1265  4          [ACT_ACL]:      BEGIN
;  956          1266  4                          ACE_POINTER = .P;
;  957          1267  4                          CASE .CODE + 1 FROM ATR$C_ADDACLENT TO ATR$C_READACE OF
;  958          1268  4                          SET
;  959          1269  4                              [ATR$C_ADDACLENT]:  0;
;  960          1270  4                              [ATR$C_DELACLENT,
;  961          1271  4                               ATR$C_MODACLENT,
;  962          1272  4                               ATR$C_DELETEACL,
;  963          1273  4                               ATR$C_FNDACLENT,
;  964          1274  4                               ATR$C_FNDACLTYP,
;  965          1275  4                               ATR$C_READACL,
;  966          1276  5                               ATR$C_READACE]:    BEGIN
;  967          1277  5                                                  CH$FILL (0, .COUNT, .ACE_POINTER);
;  968          1278  5                                                  ACE_POINTER[ACE$W_FLAGS] = SS$_ACLEMPTY;
;  969          1279  4                                                  END;
;  970          1280  5                              [ATR$C_ACLLENGTH]:  BEGIN
```

```
;  971          1281  5                                   CH$FILL (0, .COUNT, .ACE_POINTER);
;  972          1282  4                                   END;
;  973          1283  4
;  974          1284  4                           TES;
;  975          1285  3                           END;
;  976          1286  3
;  977          1287  3          TES;
;  978          1288  3
;  979          1289  2      END;                                ! end of loop
;  980          1290  2
;  981          1291  2 ! Restore the state of the protected file characteristics bits.
;  982          1292  2 !
;  983          1293  2
;  984          1294  3 HEADER[FH1$W_FILECHAR] = (.HEADER[FH1$W_FILECHAR] AND NOT PROTECTED_CHAR)
;  985          1295  2                         OR .SAVE_CHAR;
;  986          1296  2 BBLOCK [HEADER[FH1$W_RECATTR], FAT$L_HIBLK] = .SAVE_HIBLK;
;  987          1297  2
;  988          1298  2 ! Checksum the modified header.
;  989          1299  2 !
;  990          1300  2
;  991          1301  2 CHECKSUM (.HEADER);
;  992          1302  2 MARK_DIRTY (.HEADER);
;  993          1303  2
;  994          1304  2 ! Stop metering of this subfunction
;  995          1305  2 !
;  996          1306  2
;  997          1307  2 PMS_END_SUB ();
;  998          1308  2
;  999          1309  1 END;                                    ! end of routine WRITE_ATTRIB


                                                        .EXTRN  CLEANUP_FLAGS, CHECK_PROTECT
                                                        .EXTRN  CHECKSUM, MARK_DIRTY
                                                        .EXTRN  MAKE_NAMEBLOCK, GET_TIME

                                        0FFC 00000      .ENTRY  WRITE_ATTRIB, Save R2,R3,R4,R5,R6,R7,R8,R9,-; 1053
                                                                R10,R11
                                5E              34  C2 00002      SUBL2   #52, SP
                                                09  DD 00005      PUSHL   #9                                  ; 1123
                        0000G   CF              01  FB 00007      CALLS   #1, PMS_START_SUB
                                            0C  AE  D4 0000C      CLRL    CTL_ACC_GRANTED                     ; 1125
                                52          04  AC  D0 0000F      MOVL    HEADER, R2                          ; 1131
                                                52  DD 00013      PUSHL   R2
                        0000G   CF              01  FB 00015      CALLS   #1, CHECKSUM
                                                52  DD 0001A      PUSHL   R2                                  ; 1132
                        0000G   CF              01  FB 0001C      CALLS   #1, MARK_DIRTY
                        0000G   CF              02  88 00021      BISB2   #2, CLEANUP_FLAGS                   ; 1133
                                50          0C  A2  3C 00026      MOVZWL  12(R2), R0                          ; 1135
08      AE                      50 FFFF2F7F 8F  CB 0002A      BICL3   #-53377, R0, SAVE_CHAR
                          04    AE      12  A2  D0 00033      MOVL    18(R2), SAVE_HIBLK                  ; 1136
                                50  0000G   CF  D0 00038      MOVL    IO_PACKET, R0                       ; 1143
                                6E      32  A0  3C 0003D      MOVZWL  50(R0), (SP)
                                5A              04  D0 00041      MOVL    #4, I
                                            017D  31  00044      BRW     33$
                                50      08 BC4A 7E  00047 1$:   MOVAQ   @ABD[I], R0                        ; 1156
                                57              60  3C 0004C      MOVZWL  (R0), P
```

```
                                57          50  C0 0004F          ADDL2   R0, P                                   ;
                                59       02 A0  3C 00052          MOVZWL  2(R0), COUNT                            ; 1157
                                56          57  9A 00056          MOVZBL  (P)+, CODE                              ; 1158
                                            56  D7 00059          DECL    CODE
                                2F          56  D1 0005B          CMPL    CODE, #47                               ; 1165
                                            0B  14 0005E          BGTR    2$
                                        FAF1 CF46  7F 00060          PUSHAQ  ATC+4[CODE]                             ; 1168
                                50          9E  3C 00065          MOVZWL  @(SP)+, MAX_COUNT
                                50          59  D1 00068          CMPL    COUNT, MAX_COUNT                        ; 1169
                                            03  15 0006B 2$:       BLEQ    3$
                                          00A9  31 0006D          BRW     14$
                                        FADD CF46  7F 00070 3$:       PUSHAQ  ATC[CODE]                               ; 1172
                             1F             01  E1 00075          BBC     #1, @(SP)+, 4$
                                1B          0C AE  E8 00079          BLBS    CTL_ACC_GRANTED, 4$                     ; 1175
                                50          04 AC  D0 0007D          MOVL    HEADER, R0                              ; 1178
                             12 A0          04 AE  D0 00081          MOVL    SAVE_HIBLK, 18(R0)
                                      0000G CF     DD 00086          PUSHL   PRIMARY_FCB                             ; 1179
                                         04 AC     DD 0008A          PUSHL   HEADER
                                            05     DD 0008D          PUSHL   #5
                                   0000G CF        FB 0008F          CALLS   #3, CHECK_PROTECT
                                      0C AE        D0 00094          MOVL    #1, CTL_ACC_GRANTED                     ; 1180
                                        FAB8 CF46  7F 00098 4$:       PUSHAQ  ATC+3[CODE]                             ; 1187
                                52          9E  90 0009D          MOVB    @(SP)+, ACTION
                                        FAAD CF46  7F 000A0          PUSHAQ  ATC[CODE]                               ; 1188
                             02             9E  E1 000A5          BBC     #0, @(SP)+, 5$
                                            52  94 000A9          CLRB    ACTION                                  ; 1189
                                        FAA3 CF46  7F 000AB 5$:       PUSHAQ  ATC+1[CODE]                             ; 1196
                                07          00  8F 000B0          CASEB   @(SP)+, #0, #7
                  0023          001D          0016          000B4 6$:       .WORD   7$-6$,-
                  0010          0010          0010          000BC                 8$-6$,-
                                                                                  9$-6$,-
                                                                                  10$-6$,-
                                                                                  11$-6$,-
                                                                                  7$-6$,-
                                                                                  7$-6$,-
                                                                                  7$-6$,-
                                                                                  7$-6$
                                50          20 AE  9E 000C4 7$:       MOVAB   ATT_BUFFER, R0
                                            24     11 000C8          BRB     12$
                                50       0000G CF  D0 000CA 8$:       MOVL    PRIMARY_FCB, R0                         ; 1202
                                            1D     11 000CF          BRB     12$
                                50          04 AC  D0 000D1 9$:       MOVL    HEADER, R0                              ; 1203
                                            17     11 000D5          BRB     12$
                                50          04 BC  9A 000D7 10$:      MOVZBL  @HEADER, R0                             ; 1204
                                50          04 BC40 3E 000DB          MOVAW   @HEADER[R0], R0
                                            0C     11 000E0          BRB     12$
                                51          04 AC  D0 000E2 11$:      MOVL    HEADER, R1                              ; 1205
                                50          01 A1  9A 000E6          MOVZBL  1(R1), R0
                                50             6140 3E 000EA          MOVAW   (R1)[R0], R0
                                        FA61 CF46  7F 000EE 12$:      PUSHAQ  ATC+2[CODE]                             ; 1208
                                58          9E  9A 000F3          MOVZBL  @(SP)+, ADDRESS
                                58          50  C0 000F6          ADDL2   R0, ADDRESS
                             0C                 52  8F 000F9          CASEB   ACTION, #0, #12                         ; 1214
                  00C7          0092          001C          00C7 000FD 13$:      .WORD   33$-13$,-
                  0061          004A          00C7          00C7  00105               14$-13$,-
                  0098          00C7          003B          001F  0010D               25$-13$,-
                                                            00C7  00115               33$-13$,-
                                                                                      33$-13$,-
```

```
                                                                33$-13$,-
                                                                18$-13$,-
                                                                19$-13$,-
                                                                15$-13$,-
                                                                17$-13$,-
                                                                33$-13$,-
                                                                27$-13$,-
                                                                33$-13$
                        7A  11 00117          BRB     26$
                        34  BF 00119 14$:     CHMU    #52                                     ; 1224
                            04 0011B          RET
                04          59  D1 0011C 15$: CMPL    COUNT, #4                               ; 1227
                            72  1F 0011F      BLSSU   26$
                68          67  90 00121      MOVB    (P), (ADDRESS)                          ; 1230
01              A8  02      A7  90 00124      MOVB    2(P), 1(ADDRESS)                        ; 1231
                    01      A7  95 00129      TSTB    1(P)                                    ; 1232
                            05  12 0012C      BNEQ    16$
                    03      A7  95 0012E      TSTB    3(P)                                    ; 1233
                            7B  13 00131      BEQL    29$
                68          01  AE 00133 16$: MNEGW   #1, (ADDRESS)                          ; 1234
                            76  11 00136      BRB     29$                                     ; 1214
                68          67  90 00138 17$: MOVB    (P), (ADDRESS)                          ; 1239
                02          59  D1 0013B      CMPL    COUNT, #2                               ; 1240
                            7C  19 0013E      BLSS    31$
01              A8  01      A7  90 00140      MOVB    1(P), 1(ADDRESS)                        ; 1241
                            7D  11 00145      BRB     33$                                     ; 1214
                    20      AE  9F 00147 18$: PUSHAB  ATT_BUFFER                              ; 1245
                            57  DD 0014A      PUSHL   P
                            59  DD 0014C      PUSHL   COUNT
                0000G       CF  9F 0014E      PUSHAB  LOCAL_FIB
            0000G   CF          04  FB 00152      CALLS   #4, MAKE_NAMEBLOCK
        68          26  AE      0A  28 00157      MOVC3   #10, ATT_BUFFER+6, (ADDRESS)        ; 1246
                                66  11 0015C      BRB     33$                                 ; 1214
08      00              67      59  2C 0015E 19$: MOVC5   COUNT, (P), #0, #8, ATT_BUFFER     ; 1250
                                20  AE      00163
                                20  AE  9F 00165      PUSHAB  ATT_BUFFER                      ; 1251
                                14  AE  9F 00168      PUSHAB  DATBUF
                0000G       CF          02  FB 0016B      CALLS   #2, GET_TIME
            03                          56  CF 00170      CASEL   CODE, #16, #3               ; 1253
0012        000D            0008        0008      00174 20$: .WORD   21$-20$,-
                                                                21$-20$,-
                                                                22$-20$,-
                                                                23$-20$
                            59          0D  D0 0017C 21$: MOVL    #13, COUNT
                                        07  11 0017F      BRB     24$
                            59          07  D0 00181 22$: MOVL    #7, COUNT
                                        02  11 00184      BRB     24$
                                        59  D4 00186 23$: CLRL    COUNT
                68          10  AE      59  28 00188 24$: MOVC3   COUNT, DATBUF, (ADDRESS)    ; 1260
                                        35  11 0018D      BRB     33$                         ; 1214
                68          67          59  28 0018F 25$: MOVC3   COUNT, (P), (ADDRESS)      ; 1263
                                        2F  11 00193 26$: BRB     33$
                            5B          57  D0 00195 27$: MOVL    P, ACE_POINTER              ; 1266
            08              1E          56  CF 00198      CASEL   CODE, #30, #8               ; 1267
0014        0014            0014        0028      0019C 28$: .WORD   33$-28$,-
0022        0014            0014        0014      001A4      30$-28$,-
                                        0014      001AC      30$-28$,-
```

```
                                                                                30$-28$,-                                                                       ;
                                                                                30$-28$,-                                                                       ;
                                                                                30$-28$,-                                                                       ;
                                                                                30$-28$,-                                                                       ;
                                                                                32$-28$,-                                                                       ;
                                                                                30$-28$                                                                         ;
                                                        14  11 001AE 29$:       BRB     33$                                                                     ;
        59              00              6E              00  2C 001B0 30$:       MOVC5   #0, (SP), #0, COUNT, (ACE_POINTER)                                      ; 1277
                                                        6B      001B5                                                                                           ;
                        02      AB      09D0            8F  B0 001B6            MOVW    #2512, 2(ACE_POINTER)                                                   ; 1278
                                                        06  11 001BC 31$:       BRB     33$                                                                     ; 1267
        59              00              6E              00  2C 001BE 32$:       MOVC5   #0, (SP), #0, COUNT, (ACE_POINTER)                                      ; 1281
                                                        6B      001C3                                                                                           ;
                        02              5A              6E  F2 001C4 33$:       AOBLSS  (SP), I, 34$                                                            ; 1143
                                                        03  11 001C8            BRB     35$                                                                     ;
                                                    FE7A    31 001CA 34$:       BRW     1$                                                                      ;
                                        52      04      AC  D0 001CD 35$:       MOVL    HEADER, R2                                                              ; 1294
                                        50      0C      A2  3C 001D1            MOVZWL  12(R2), R0                                                              ;
                                        50  0000D080     8F  CA 001D5            BICL2   #53376, R0                                                              ;
                    0C  A2              50      08      AE  A9 001DC            BISW3   SAVE_CHAR, R0, 12(R2)                                                   ; 1295
                        12  A2          04      AE  D0 001E2            MOVL    SAVE_HIBLK, 18(R2)                                                              ; 1296
                                                        52  DD 001E7            PUSHL   R2                                                                      ; 1301
                    0000G   CF                      01  FB 001E9            CALLS   #1, CHECKSUM                                                            ;
                                                        52  DD 001EE            PUSHL   R2                                                                      ; 1302
                    0000G   CF                      01  FB 001F0            CALLS   #1, MARK_DIRTY                                                          ;
                    0000G   CF                      00  FB 001F5            CALLS   #0, PMS_END_SUB                                                         ; 1307
                                                        04  001FA            RET                                                                             ; 1309

; Routine Size:  507 bytes,    Routine Base:  $CODE$ + 04AE

; 1000          1310  1
; 1001          1311  1 END
; 1002          1312  0 ELUDOM


;                                       PSECT SUMMARY
;
;       Name                            Bytes                           Attributes
;
;   $CODE$                                1705  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $LOCKEDD1$                             512  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                                   Library Statistics
;
;                                           -------- Symbols --------      Pages       Processing
;       File                                Total    Loaded   Percent      Mapped      Time
;
;   _$255$DUA28:[SYSLIB]LIB.L32;1            18619       103         0        1000        00:02.0
```

```
;                                       COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:RWATTR/OBJ=OBJ$:RWATTR MSRC$:RWATTR/UPDATE=(ENH$:RWATTR)

; Size:         1316 code + 901 data bytes
; Run Time:        00:35.1
; Elapsed Time:    01:10.5
; Lines/CPU Min:    2241
; Lexemes/CPU-Min: 19337
; Memory Used:  256 pages
; Compilation Complete
```

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

REQUEU
LIS

RWATTR
LIS

MODIFY
LIS

SCHFCB
LIS

MAKAEC
LIS

MPWIND
LIS

MAPVBN
LIS

PMS
LIS

RDHEDR
LIS

RWVB
LIS

RETDIR
LIS

RDBLOK
LIS

SMALOC
LIS

MOUNT
LIS

MAKNMB
LIS

MAKSTR
LIS

MXTHDR
LIS